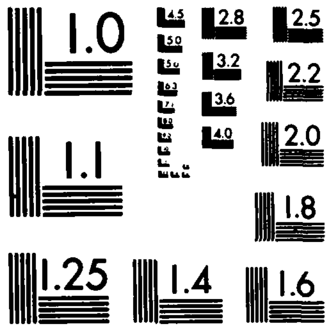
AD-A136 812 SIMTBED: A GRAPHICAL TEST BED FOR ANALYZING AND REPORTING THE RESULTS OF A STATISTICAL SIMULATION EXPERIMENT(U) NAVAL POSTGRADUATE SCHOOL MONTEREY CA 1/2

UNCLASSIFIED H W DRUEG SEP 83 F/G 9/2 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

NAVAL POSTGRADUATE SCHOOL

Monterey, California

AD-A136812

THESIS

SIMTBED a Graphical Test Bed for Analyzing
and Reporting the Results of a
Statistical Simulation Experiment

by

Hans-Walter Drueg

September 1983

Thesis Advisor: P.A.W. Lewis

Approved for public release; distribution unlimited.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. AD-A136812	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) SIMTBED a Graphical Test Bed for Analyzing and Reporting the Results of a Statistical Simulation Experiment		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; September 1983
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Hans-Walter Drueg		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		12. REPORT DATE September 1983
		13. NUMBER OF PAGES 158
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

Accession For
NTIS GRA&I
DTIC TAB
Unannounced
Justification
By
Distribution/
Availability Codes
Avail and/or
Dist Special
A1

DTIC COPY INSPECTED

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

SIMTBED
Statistical Simulation
Graphics
Regression-Adjusted Graphics
Serial Correlation Coefficient
Gamma Distribution
Standard Deviation of Correlation Coefficient

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A graphical test bed in which the results of a simulation experiment can be reported and analyzed is described. The test bed is based on the regression adjusted graphics and estimation methodology developed by Heidelberger and Lewis [Ref. 1] for regenerative simulation. From the graphics and associated numerics, the experimenter can summarize and see simultaneously relative properties, such as bias, normality and standard

DD FORM 1473, 1 JAN 73 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

#20 - ABSTRACT - (CONTINUED)

deviation, of several estimators of a characteristic of a population for up to 8 sample sizes. The evolution of these properties with sample size is also displayed. The graphics is supported on a line printer to make it and the program portable. The technique is illustrated by examples concerning the effects of changes in data distribution on the behavior of the lag one serial correlation coefficient, the estimation of the shape parameter of Gamma random variables and a comparison of different methods (jackknife, bootstrap) for estimating the standard error of an estimator.

Approved for public release; distribution unlimited.

SIMTBED a Graphical Test Bed for Analyzing and Reporting the Results of a Statistical Simulation Experiment

by

Hans-Walter Drueg
Major, German Air Force
Ing.(grad.), Hochschule der Bundeswehr Muenchen, 1974

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
September 1983

Author: Hans-Walter Drueg

Approved by: Peter A.W. Lewis
Thesis Advisor

Second Reader

Chairman, Department of Operations Research

K.T. Marshall
Dean of Information and Policy Sciences

ABSTRACT

A graphical test bed in which the results of a simulation experiment can be reported and analyzed is described. The test bed is based on the regression adjusted graphics and estimation methodology developed by Heidelberger and Lewis [Ref. 1] for regenerative simulation. From the graphics and associated numerics, the experimenter can summarize and see simultaneously relative properties, such as bias, normality and standard deviation, of several estimators of a characteristic of a population for up to 8 sample sizes. The evolution of these properties with sample size is also displayed. The graphics is supported on a line printer to make it and the program portable. The technique is illustrated by examples concerning the effects of changes in data distribution on the behavior of the lag one serial correlation coefficient, the estimation of the shape parameter of Gamma random variables and a comparison of different methods (jackknife, bootstrap) for estimating the standard error of an estimator.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGEMENTS

I wish to acknowledge three people who contributed to my thesis research. First is Professor P.A.W. Lewis who was my thesis advisor. His time and his advice was always available. His support contributed a great deal to the increase of my knowledge and the completion of this the 's for which I am grateful.

Second I would like to thank Professor E.J. Orav wh was always available to answer questions and to discuss unsolved problems. His advice was especially valuable to me in running and interpreting the output of the example program runs.

Third, but not last, I wish to thank L. Uribe who had always time and patience to discuss programming problems. He contributed the "SIMTB3" FORTRAN program which is part of this thesis.

I. SYNOPSIS

SIMTBED

THE PROGRAM:

Portable FORTRAN program using printer plot graphics

(3 different program versions)

Program will run on:

IBM

VAX

IBM PC

etc.

ca. 900 lines of FORTRAN Code

Memory requirements:

SIMTB1	1 M Bytes
SIMTB2	1 M Bytes
SIMTB3	0.5 M Bytes

(may slightly differ with different type of estimator functions and subsample sizes)

PURPOSES:

- To explore the distribution of a statistical estimator
- To see how that distribution changes with sample size
- To compare that distribution with the distribution of competing estimators

THE USER SUPPLIES:

A. Optional Parameters:

NE(1),NE(2),...,NE(8) = Subsample Sizes (maximum is 8)

The estimator is computed based on NE(i) data points

N = Total Number of simulated data points per replication

At Subsample Size NE(i), there are $\lfloor N/NE(i) \rfloor$ independent values of the estimator

M = Number of Replications

When all replications have been run, there are $M^* \lfloor N/NE(i) \rfloor$ independent values of the estimator at each NE(i)

D = Degree of Regression (maximum D = 3)

L = Number of Subsample Sizes (maximum L = 8)

GRAPHICS and SCALING options

B. Data:

A total of M*N simulated data values are needed. In SIMTB1 and SIMTB2 the same data is used at each subsample size (NE(i)). In SIMTB3 new data is always generated.

C. Estimators:

Up to 3 FORTRAN functions (i.e. Estimators) are needed. They must accept as inputs a data subsample and the size of that subsample. They must return one value of the estimator for that subsample.

SIMTBED PRODUCES:

- A one page graphical output (box plots) at each subsample size
- Numerical Summaries at each subsample size (mean, Std. Dev., Std. Dev. of the mean, skewness, kurtosis)
- Regressions to quantify changes in the mean and variance as subsample size changes

II. INTRODUCTION

SIMTBED, with the different versions (SIMTB1, SIMTB2 and SIMTB3) is a graphical display program. The program is based on the program RAGE [Ref. 2]. It is used, with simulated data, on a digital computer. The program can be used to examine statistical estimators of different type, or properties of a single estimator under different distributional assumptions. The distribution of the estimator can be explored for given sample sizes and the properties can be compared for different sample sizes. The estimation conditions are controlled by the experimenter. It is also possible to examine the effects of changes in the underlying distribution of the data.

When the program SIMTB1 or SIMTB2 is used with simulated data, the data is assumed to be independent and identically distributed (iid). This iid data can be sectioned into M independent blocks of specified sample size N. The sample of size N, will be sectioned into smaller subsample of size NE(k). The estimates are then calculated from this subsample of size NE(k).

One salient feature of the program versions SIMTB1 and SIMTB2 is that they use the same batch of simulated random variables to explore the properties of all the estimators at the various subsample sizes. This is done for economy

and could be important on slow computers; the price paid is that the regression analysis provided by SIMTB1 and SIMTB2 of its graphical output is performed on correlated samples.

The version SIMTB3 uses one dimensional data and does not have this repetition feature. New data is used for each calculation of each estimator at each subsample size. The data is generated when the estimator function is called and only the needed batch of the exact subsample size is generated. This technique reduces the memory requirements.

Moreover all data sets are uncorrelated and a much more precise correlation can be performed if required. But this technique increases the computer time.

To use the program it is necessary only to define the optional parameters (see Section IV), supply the simulated random variables or a batch of data points, and provide the FORTRAN functions for the calculation of the estimators which are of interest. The program itself will subdivide the input data and feed the data properly into the functions, scale the graphic display, produce boxplots and summary statistics. A regression will be computed for the mean and variance of each estimator based on inverse subsample size. The result of the regression is displayed graphically and numerically.

The program is written in ANSI Standard FORTRAN (X3.9-1966) and extensively tested on an IBM 3033 computer using FORTRAN IV (H Extended) or FORTRAN IV (G1) compilers. The

program SIMTBED (all versions) provides all subroutines used inside the program. Besides the estimator functions, the user has to provide NO additional subroutines via packages like IMSL. The program should be totally portable; it has been tested under FORTRAN 77 on a VAX 11/780. The only limitation is the available memory.

III. GENERAL IDEA AND DATA STRUCTURE OF THE PROGRAM

The main purpose of SIMTBED, with the different versions, is to explore the distributional behavior of estimators and show their properties in a graphical and numerical display. All versions use the same ideas, they only differ in the type of data and the way the data is used inside the program.

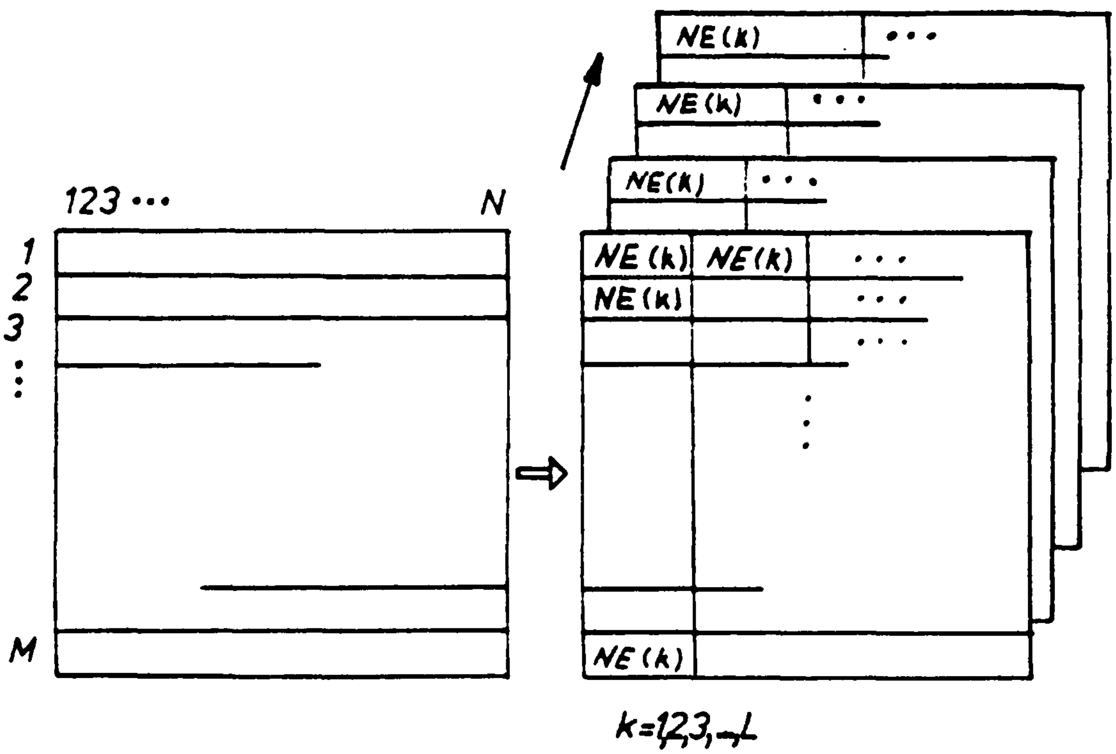


Figure 1. Sectioning of the Data into M*N Sections

To study the behavior of an estimator the experimenter usually uses well known simulated data. Thus if one is interested in exploring the behavior of estimates of the shape parameter in a Gamma population, one generates Gamma variates from a random number generator package (e.g., IMSL subroutine Chap. G). This batch of simulated data is processed by SIMTBED in the following way.

The data batch is first divided into M independent blocks. Each block contains N data points. So the starting data batch has to consist of M*N data points (see Figure 1).

All blocks are divided into subsamples of size n_i. The actual subsample size n_i is an element of the subsample size array NE. This array can store up to 8 different values. Then the estimator is calculated for each subsample of size n_i. The estimator function will be calculated $(\lfloor N/n_i \rfloor)$*M times. This total population of estimates is used to evaluate the summary statistics for the estimate and construct the corresponding box plot. If NE contains another element, the blocks are divided into the new subsample n_{i+1} size and all calculations are done again.

In addition to the summary statistics and the box plots (see e.g., Figure 3a), a regression on the averages and on the variance is computed, following the methodology of regression adjusted estimate (RARE) developed by Heidelberger and Lewis [Ref. 1].

The RARE estimate is the regression coefficient α_0. It is the asymptotic estimate of the expected value of the parameter. The unbiased RAGE estimate of the average of the parameter is determined by the regression formula:

$$E(\theta(n_i)) = \alpha_0 + \alpha_1 \frac{1}{n_i} + \alpha_2 \frac{1}{n_i^2} + \ldots + \alpha_D \frac{1}{n_i^D}$$

where D is an input parameter.

The asymptotic RARE estimate is printed as a dashed line in the graph.

The unbiased RAGE estimate of the variance of the estimator is given by the formula:

$$S^2(n_i) = \beta_1 \frac{1}{n_i} + \beta_2 \frac{1}{(n_i)^{1.5}} + \beta_3 \frac{1}{(n_i)^2} + \ldots$$

The regression calculations of the average are done for each block separately. The result are M sets of regression coefficients. The finally printed regression coefficients are the averages of the M replications (see Figure 2). From the set of coefficients the variance and the standard deviation of the regression coefficients is calculated.

The regression on the variance is done once, using all (M*N) data points.

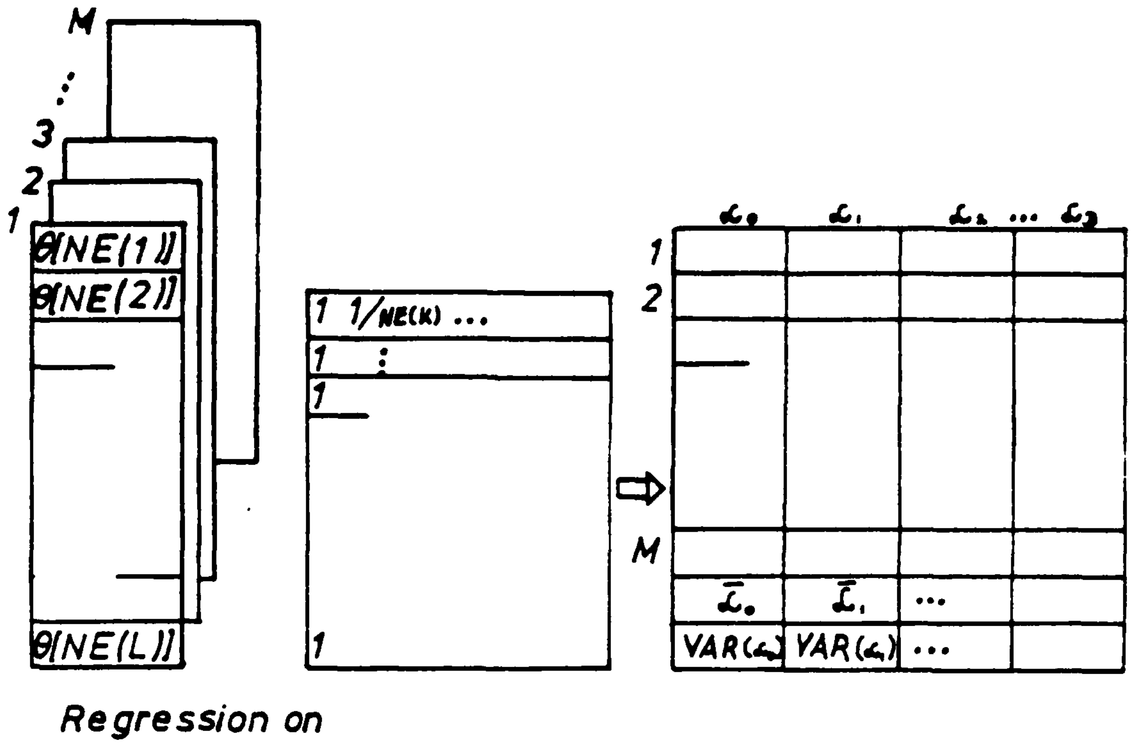


Figure 2. Structure of the Regression Coefficients

IV. ARGUMENTS OF THE PROGRAMS SIMTB1 AND SIMTB2

All versions of SIMTBED have in general the same argument list. SIMTB2 (a version for multivariate random variables) has two additional arguments. All arguments will be described in detail and all restrictions or limitations will be discussed. In Section V, detailed examples of setups for SIMTB1 and SIMTB2 are given.

A. X--DATA ARRAY

The array X is the input data array. X is single precision REAL*4, and is generally simulated data e.g., in Section VII, independent Gamma variates.

In SIMTB1 the dimension of the array X must not exceed 50,000.

In SIMTB2 the input data is multivariate and consequently the array X has two dimensions. The size of the dimensions is not directly limited by the program. The space must be provided by the calling program and is passed as an argument (see IR and IRK). The memory requirements increase rapidly as the dimensions of X increase.

B. N--SAMPLE SIZE

The sample size N is the number of data points per section of input data X. N is an INTEGER. Depending on the precision of the simulation that is required, the sample size N can vary from 1 to 50,000.

In SIMTB2 N is the number of multivariate data points per section. M (the number of replications) times N must not exceed IR, the row dimension of X.

If M times N exceeds 50,000 an error message will be printed and the execution will be terminated. If the product M times N exceeds the total number of data points provided by the user, NO error message will signal the user error and the result of the execution is not predictable.

C. M--NUMBER OF REPLICATIONS

The number of replications M of the array X is an INTEGER. M determines the number of sections, into which the data set X is divided. So M*N is the dimension of X in SIMTB1. The parameter M also determines the number of regressions on the mean values that will be run to find the regression coefficients in the regression on the mean. If M is 1 only one regression will be done and no variance and standard deviation of the regression coefficients can be calculated.

D. NE--SUBSAMPLE SIZE ARRAY

The argument NE is an INTEGER array, containing up to 8 subsample sizes. These are the subsample sizes at which the properties of the estimator are to be investigated. NE must always contain 8 elements. If less than 8 subsample sizes are used, the array must be filled up with dummy values. The estimator is calculated for each subsample

individually. The used values of NE (not the dummy values) have to be ordered from the smallest to the largest (NE(1) < NE(2) < NE(3) < ...).

The program can only handle up to a total number of 12,500 estimates. This limit is caused by storage limitations and can be expanded for larger computers. The smallest subsample size NE(1) produces the most estimates and if this case exceeds 12,500 the execution will be terminated and an error message will be printed.

In SIMTB2 all values of NE have to be less than 5,000.

E. L--NUMBER OF SUBSAMPLE SIZES (BOX PLOTS)

The number of box plots L in the graphical output (the number of subsamples at which the estimator is examined), is an INTEGER with values from 1 to 8. The value of L determines how many elements of the array NE the program will use (e.g., L = 2, the program uses only NE(1) and NE(2) for the calculations). L determines the number of box plots and the number of corresponding summary statistics in the output. If L is out of range the program will terminate execution and print an error message.

For L = 1 no regression is possible. No regression output will be printed.

F. D--DEGREE OF REGRESSION

The degree of regression on the mean D, is an INTEGER with values from 1 to 3. The chosen degree refers to the

number of coefficients calculated and printed for the regression equations. Experience has shown that D = 3 is preferable (higher values cause severe numerical problems in the regression computations).

If D exceeds the value of L-1, the program reduces D to this number, regardless of the value chosen by the user.

G. RG--REDUCED GRAPHICS

The argument RG is an INTEGER with value:

0 = off ==> full graphics

1 = set ==> reduced graphics

In the reduced graphics case the vertical scale of the graph is reduced by eliminating the outliers. Only the number of outliers is counted and the number is printed. This enables the user to see the body of the boxplot in more detail.

H. SEI--SCALING ESTIMATORS INDIVIDUALLY

The argument SEI is an INTEGER with value:

0 = off ==> common scaling

1 = set ==> individual scaling.

For SEI = 0 all printed graphs (max. 3 per program run) are scaled to the same range. This makes the comparison of the different graphs easier. The value SEI = 1 causes the program to scale each graph individually.

The argument SEI works in combination with the arguments RG and SVS. The combination of these three arguments make it possible to fit the printed graphs to the needs of the user.

I. SVS--SETTING THE VERTICAL SCALE

The argument SVS is an INTEGER with value:

0 = off ==> automatic scaling

1 = set ==> scaling by the user.

SVS = 0 causes the program to calculate the scale of the graphic printout. The final graphic display is influenced by the chosen values of RG and SEI. The values of YMIN and YMAX are ignored by the program.

For SVS = 1 the user must provide the scaling. The graphics are vertically scaled to a given minimum value (YMIN) and maximum value (YMIN and YMAX). The arguments of RG and SEI are ignored. Only the numbers of outliers outside the display are printed for each box plot.

J. YMIN--MINIMUM VALUE OF THE VERTICAL SCALE

The scaling parameter YMIN is data type REAL*4. It is the lower limit of the vertical scale. It affects the scaling only in combination with SVS = 1. If the chosen value is so large that the value YMIN lies inside the body of the box plot, an error occurs and the program ends with an abnormal ending.

K. YMAX--MAXIMUM VALUE OF THE VERTICAL SCALE

The scaling parameter YMAX is data type REAL*4. It is the upper limit of the vertical scale. The scaling is only effected if SVS has value 1. If the value of YMAX is too small and it lies inside the body of the box plot, the program comes to an abnormal ending.

YMIN and YMAX in combination with SVS = 1 should only be used for well known graphic ouptut. With this option, it is possible to scale the output of different program runs to a common scale. In particular if more than 3 estimators have to be estimated and compared, so that a common scale is needed, this option may be used.

L. NEST--NUMBER OF ESTIMATORS

The parameter NEST is an INTEGER with the value 1, 2 or 3. The value of NEST determines the number of different one-page graphic displays the program produces with one call from the calling program. Usually the value of NEST is equal to the number of different estimators used in the program.

In SIMTB2 the same estimator may be used with different (e.g., normal, exponential etc.) distributed data sets.

M. EST1, EST2, EST3--ESTIMATOR FUNCTIONS

These 3 parameters are data type REAL*4 and are used to pass the EXTERNAL function names to the program. An external declared function is a function which computes the value of an estimator. This function may call other routines, but the final value of the estimator must be passed over this function name.

For SIMTB1 each function must have the two arguments X and NEK (e.g., FUNCTION VARIANCE (X,NEK)). X is the data array and NEK is the number of data elements in X.

SIMTB2 needs for each function four arguments X, NEK, IDR, IRK (e.g., FUNCTION CORRELATION (X,NEK,IDR,IRK)). X is a two dimensional data array and NEK is the subsample size, for which the function is evaluated. IDR and IRK are the dimensions of the array X.

If the user wants to use less than 3 estimator functions, he must use dummy arguments and choose the correct value of NEST. The easiest way to do this is to use a function of a previous used estimator again (for details of the programming see Section V).

N. TTL1, TTL2, TTL3--DESCRIPTION OF THE ESTIMATORS

These arrays are used to pass titles from the calling program to SIMTB. Each title has to be 120 characters long (15 fields, each 8 characters wide). The title is printed below the output of the corresponding function.

If the user uses less than 3 functions and titles, he has to use dummy arguments (for details see Section V).

If the titles are not in the correct format, corresponding to the FORMAT statement the program will not execute properly.

O. IR, IRK--DIMENSIONS OF THE ARRAY X

These arguments are used ONLY in SIMTB2. IR (row dimension) and IRK (column dimension) are INTEGERS and have to match the dimension of X in the calling program.

If the dimensions don't match, NO error message will be produced and the output is unpredictable. The error may not be obvious, and the ouput may look reasonable.

V. DUMMY EXAMPLES OF IMPLEMENTING SIMTB AND SIMTB2 INTO A DRIVER-PROGRAM

The following examples show how the programs SIMTB1 or SIMTB2 can be implemented into a FORTRAN driver-program. The only purpose of the examples is to clarify the FORTRAN implementation, to avoid programming errors by the user. Additional comment lines are added to the program examples.

In the first example SIMTB1 is used to compare two different estimators (VAR and UNBVAR) of the variance of a normal sample. The sample may be generated with a random number generator (e.g., LLRANDOMII).

In the second example SIMTB2 is used to compare the distribution of two estimators. The estimators are the covariance (COV) and the correlation coefficient (CORR) of a bivariate standard normal sample.

A. EXAMPLE 1 USING SIMTB1

```
      MAIN
C   EXAMPLE of SIMTB1   Calling program, it has not to be
C   the MAIN program
      REAL*4 X(50000), YMIN, YMAX, VAR, UNBVAR
      REAL*8 T1(15), T2(15), T3(15)
      INTEGER N, M, NE(8), L, D, RG, SEI, SVS, NEST
C
      DATA N   /2500/
```

```
      DATA M   /  20/
      DATA NE  /10,20,25,50,100,0,0,0/
C  Array NE must have 8 elements, if only 5 are used, add
C  dummy variables and set L = 5
      DATA L   /   5/
      DATA RG  /   0/
      DATA SEI /   0/
      DATA SVS /   0/
      DATA NEST/   2/
      DATA T1  /'ESTIMATE','OF THE V','ARIANCE','USING  ',
     +'VAR=(1/N)','*SUM(X(I',')-XBAR)*','*2      ',7*' '/
      DATA T2 /'ESTIMATE','OF THE V','ARIANCE ','USING  ',
     +'VAR=(1/(1','-N))*SUM','(X(I)-XB','AR)**2  ',7*' '/
C  All 15 fields (each 8 characters) have to be
C  initialized
C
      EXTERNAL VAR, UNBVAR
C
C  Generate M*N independent Normal (0,1) Random numbers
C  and store into X
C
      CALL SIMTB1(X,N,M,NE,L,D,RG,SEI,SVS,YMIN,YMAX,NEST
     +           ,VAR,T1,UNBVAR,T2,VAR,T1)
C  EST3=VAR and TTL3=T1 used as dummy variables
C
      STOP
      END
```

```
C
      FUNCTION VAR (X,NEK)
C  Function to calculate the Variance.
C  All calculations inside the function shculd be done in
C  DOUBLE PRECISION
C
      REAL*4 X(N), VAR
      REAL*8 SUM, XBAR, DVAR
C
      SUM=0.0D0
      DO 10 I=1,N
         SUM=SUM+DBLE(X(I))
      10 CONTINUE
         XBAR=SUM/FLOAT(N)
         SUM=0.0D0
      DO 20 I=1,N
         SUM=SUM+((DBLE(X(I)))-XBAR)**2
      20 CONTINUE
         DVAR=SUM/FLOAT(N)
         VAR=SNGL(DVAR)
C
         RETURN
         END

         FUNCTION UNBVAR (X,NEK)
C  Function to calculate the Variance.
C  All calculations inside the function should be done in
```

```
C  DOUBLE PRECISION
C
      REAL*4 X(N), UNBVAR
      REAL*8 SUM, XBAR, DUNVAR
C
      SUM=0.0D0
      DO 10 I=1,N
         SUM=SUM+DBLE(X(I))
 10   CONTINUE
      XBAR=SUM/FLOAT(N)
      SUM=0.0D0
      DO 20 I=1,N
         SUM=SUM+((DBLE(X(I)))-XBAR)**2
 20   CONTINUE
      DUNVAR=SUM/FLOAT(N)
      UNBVAR=SNGL(DUNVAR)
C
      RETURN
      END
```

B. EXAMPLE 2 USING SIMTB2

```
     MAIN
C  EXAMPLE of SIMTB2   Calling program, it has not to be
   the MAIN program
     REAL*4 X(25000,2), YMIN, YMAX, COV, CORR
     REAL*8 T1(15), T2(15), T3(15)
```

```
      INTEGER N, M, NE(8), L, D, RG, SEI, SVS, NEST, IR,
     +        IRK
C
      DATA N   /2500/
      DATA M   /  10/
      DATA IR  /25000/
      DATA IRK /   2/
C  IR and IRK must be equal to the dimensions of X
      DATA NE  /10,20,25,50,83,100,125,250/
      DATA L   /   8/
      DATA RG  /   0/
      DATA SEI /   0/
      DATA SVS /   1/
      DATA YMIN/ 0.0/
      DATA YMAX/ 1.0/
      DATA NEST/   2/
      DATA T1  /'ESTIMATE','OF THE C','OVARIANC','E       ',
     +11*'  '/
      DATA T2 /'ESTIMATE','OF THE C','ORRELAT1','ON COEFF',
     +'ICIENT  ','10*' '/
C  All 15 fields (each 8 characters) have to be
C  initialized
C
     EXTERNAL COV, CORR
C
C  Generate M*N pairs of independent random bivariate
C  numbers, each pair being independent N(0,1) random
```

```
C   variables and store into X
C
      CALL SIMTB2 (X,N,M,NE,L,D,RG,SEI,SVS,YMIN,YMAX,NEXT
     +            ,COV,T1,CORR,T2,COV,T1,IR,IRK)
C   EST3=COV and TTL3=T1 used as dummy variables
C
      STOP
      END
C
C
      FUNCTION COV (X,NEK,IDR,IRK)
C   Function to calculate the Covariance.
C   All calculations inside the function should be done in
C   DOUBLE PRECISION
C
      REAL*4 X(IDR,IRK), COV
      REAL*8 SUM1,SUM2,SUM3,XBAR1,XBAR2,EX1X2,DCOV
C
      SUM1=0.0D0
      SUM2=0.0D0
      SUM3=0.0D0
      DO 10 I=1,N
         SUM1=SUM1+DBLE(X(I,1))
         SUM2=SUM2+DBLE(X(I,2))
         SUM3=SUM3+DBLE(X(I,1)*X(I,2))
   10 CONTINUE
      XBAR1=SUM1/FLOAT(N)
```

```
      XBAR2=SUM2/FLOAT(N)
      EX1X2=SUM3/FLOAT(N)
      DCOV=EX1X2-(XBAR1*XBAR2)
      COV=SNGL(DCOV)
C
      RETURN
      END

C
      FUNCTION CORR (X,NEK,IDR,IRK)
C  Function to calculate the Correlation coefficient
C  All calculations inside the function should be done in
C  DOUBLE PRECISION
C
      REAL*4 X(IDR,IRK), CORR
      REAL*8 SUM1,SUM2,SUM3,XBAR1,XBAR2,EX1X2,VAR1,VAR2,
     +       COV,DCORR
C
      SUM1=0.0D0
      SUM2=0.0D0
      SUM3=0.0D0
      DO 10 I=1,N
         SUM1=SUM1+DBLE(X(I,1))
         SUM2=SUM2+DBLE(X(I,2))
         SUM3=SUM3+DBLE(X(I,1)*X(I,2))
   10 CONTINUE
      XBAR1=SUM1/FLOAT(N)
```

```
      XBAR2-SUM2/FLOAT(N)
      EX1X2=SUM3/FLOAT(N)
      SUM1=0.0D0
      SUM2=0.0D0
      DO 20 I=1,N
         SUM1=SUM1+DBLE(X(I,1)**2)
         SUM2=SUM2+DBLE(X(I,2)**2)
 20   CONTINUE
      VAR1=(SUM1/FLOAT(N))-(XBAR1**2)
      VAR2=(SUM2/FLOAT(N))-(XBAR2**2)
      COV=EX1X2-(XBAR1*XBAR2)
      DCORR=COV/((VAR1*VAR2)**0.5)
      CORR=SNGL(DCORR)
C
      RETURN
      END
```

VI. STUDY OF THE BEHAVIOR OF SERIAL CORRELATION ESTIMATES FOR DIFFERENT DISTRIBUTIONS

A. CALCULATION OF THE FIRST SERIAL CORRELATION COEFFICIENT

It is known that for an independent sample from a population with finite variance, the distribution of the serial correlation coefficient (Anderson and Walker, 1964) [Ref. 3] is asymptotically Normal with mean zero and variances $1/n$, where n is the sample size. If the population is i.i.d Normal then the bias is exactly $-1/n$. Since those asymptotic properties are frequently used as approximations in tests of significance, it is important to know how valid the approximation would be in small samples from a variety of distributions. We will look at that question in the next two sections and then go on to consider two alternative measures of correlation, Fisher's z-transform and the 2-fold jackknifed estimate of the correlation. Their ability to reduce bias and/or induce Normality will be examined against other changes in the distribution of the estimators, particularly variance inflation. A simulation study, without graphics, of some of these problems was conducted by Cox (1966) [Ref. 4].

B. SIMTB1 OUTPUT FOR SERIAL CORRELATION

Figure 3(a) shows the simulated distribution and sample properties of the serial correlation coefficient estimate

$$r_n = \frac{n \sum_{j=1}^{n-1} (X_j - \bar{X}_1)(X_{j+1} - \bar{X}_n)}{(n-1) \sum_{j=1}^{n} (X_j - \bar{X}_0)^2},$$

where:

$$\bar{X}_0 = \sum_{j=1}^{n} X_j/n,$$

$$\bar{X}_1 = \sum_{j=1}^{n-1} X_j/(n-1), \text{ and}$$

$$\bar{X}_n = \sum_{j=2}^{n} X_j/(n-1)$$

for various sub-sample sizes $n = n_i$. This definition matches that used by Anderson and Walker (1964). We consider first subsamples of size $n_1 = 10$, and then of size $n_2 = 20$, $n_3 = 30$, $n_4 = 40$, $n_5 = 50$, $n_6 = 75$, $n_7 = 100$ and $n_8 = 150$, successively. For each subsample size the input sample of $N = 5000$ simulated Normal (0,1) random variables is divided into as many full subsamples of size n_i as possible, and the serial correlation is computed for each of the $\lfloor N/n_i \rfloor$ subsamples of size n_i. The entire procedure is then replicated $M = 10$ times, each time with a new simulated sample of $N = 5000$ Normal (0,1) variables.

After all M replications have been run, all the estimates of serial correlation for each subsample size are grouped together and their simulated distribution is presented via a

boxplot and summary statistics. The boxplot follows the standards discussed in Mosteller and Tukey (1977) [Ref. 5] with the median denoted by a + within the box, the mean by a * within the box, the outliers by 0's, and the far outliers by *'s beyond the whiskers. The summary statistics include the sample mean, sample standard deviation, estimated standard deviation of the sample mean (i.e., sample standard deviation/ sqrt($M \lfloor N/n_i \rfloor$), sample skewness and sample kurtosis of the correlation estimates.

Looking at the output, the first (leftmost) boxplot in the graph in Figure 3(a) shows the distribution of

$$(\#\ \text{Replications}) \times \left\lfloor \frac{(\text{Simulation Sample Size})}{(\text{Subsample Size})} \right\rfloor = 10 \times \left\lfloor \frac{5000}{10} \right\rfloor$$

$$= 10 \times 500 = 5000$$

estimates of serial correlation from independent subsamples of size $n_1 = 10$. Summary statistics for the boxplot can be found below the graph in the column labeled "Subsample Size 10," so that the average serial correlation is -.1074, and the estimated standard deviation is .2996. The estimated standard deviation of the serial correlation estimate is $.2996/\sqrt{(5000)} = .00424$. Recall that this refers to correlation estimates based on subsamples of size 10.

Since the X-axis of the graph represents subsample size, the last (rightmost) boxplot shows the distribution of

$$10 \times \left\lfloor \frac{5000}{150} \right\rfloor = 10 \times 33 = 330$$

estimates of serial correlation from independent subsamples of size $n_8 = 150$. Although the 330 estimates are independent of each other, they are not independent of the 5000 estimates that comprise the first boxplot since the same data (divided and processed in different ways) was used for both. Summary statistics show that the average correlation has dropped to -.007372, indicating the fall off in bias, and the standard deviation has dropped to .07822, indicating the greater accuracy with which the correlation can be estimated when 150 points, rather than 10, are available.

In order to quantify the changes that are occurring in the mean and variance of the distribution of the estimator as subsample size changes, SIMTB1 performs two types of regressions. The first regression is on the averages and is done after each replication, using the average serial correlation for that replication, $\bar{r}_{n_i}$, as the dependent variable. Inverse powers of the subsample size serve as the independent variables. For Figure 3(a) the degree of the regression was chosen to be D = 3 so, for each replication, the equations we attempt to fit by least squares are:

$$\bar{r}_{n_i} = a_0 + \frac{a_1}{n_i} + \frac{a_2}{n_i^2} + \frac{a_3}{n_i^3} \quad \text{for} \quad i = 1,2,\ldots,8.$$

This form anticipates the general asymptotic expansion

$$E(\hat{\Theta}(n)) \quad = \quad \Theta + \frac{\alpha_1}{n} + \frac{\alpha_2}{n^2} + \frac{\alpha_3}{n^3} + \ldots$$

which holds true in the current situation with $\Theta = 0$ and (in the Normal case) $\alpha = -1$ (see Cramer (1948) for general results of this type) [Ref. 6].

Values of a_0, a_1, a_2, and a_3 are calculated after each replication, averaged across the M replications to get $\bar{a}_0$, $\bar{a}_1$, $\bar{a}_2$, and $\bar{a}_3$, and then the averages are reported below the summary statistics on the line "Mean of Regression on Averages--Coefficients." We find that $\bar{a}_0 = -.000272$ and $\bar{a}_1 = -1.03074$, both close to their theoretical counterparts.

Because we have 10 replications and therefore 10 independent values of each of a_0, a_1, a_2, and a_3, we can also estimate the variances and standard deviations of a_0, a_1, a_2, and a_3 across replications. These values are presented on the two lines immediately below the coefficients. For instance, the estimated s.d. of the estimate $\bar{a}_0 = -.000272$ of a_0 is .003892.

The regression line for the mean value of the estimator is presented visually in the graph as a dotted curve. The estimated asymptote (i.e., $\bar{a}_0$) is printed with a dashed line wherever it does not coincide with the regression line. Bias, therefore, can be viewed as the difference between those two lines.

The second regression referred to above is done after all replications have been run and the variances of the estimators

at each subsample size have been calculated. (Note that the standard deviations, not the variances, are presented in the summary statistics.) It should be recalled from previous discussion that these variances, as well as all measures in the summary statistics, are based on the grouping together of the serial correlations from all replications, at each subsample size. This is in contrast to the procedure for the regression on the means, where average correlations are computed for each subsample size for each replication. In the case of the variances, we have 8 equations:

$$\hat{\mathrm{Var}}(r_{n_i}) = \frac{b_0}{n_i} + \frac{b_1}{n_i^{3/2}} + \frac{b_2}{n_i^2} + \frac{b_3}{n_i^{5/2}}, \quad i = 1,2,\ldots,8,$$

which we fit by least squares in order to estimate the coefficient β_0, β_1, β_2, and β_3 in the presumed asymptotic expansion

$$\mathrm{Var}(\hat{\Theta}(n)) = \frac{\beta_0}{n} + \frac{\beta_1}{n^{3/2}} + \frac{\beta_2}{n^2} + \frac{\beta_3}{n^{5/2}} + \cdots$$

This expansion holds for the variance of the estimated serial correlation coefficient for independent data. Usually it will be β_0 in which we are most interested since β_0 is used in computing asymptotic relative efficiencies of estimators. For independent data with finite variance, we know that $\beta_0 = 1$. The computed values of b_0, b_1, b_2, and b_3, are presented on the line labeled 'Regression on Variance--Coefficients'. Notice that $b_0 = .7438$ is close to the theoretical value of 1.

The final two numbers on Figure 3(a), YMIN and YMAX, simply show the scale of the vertical axis. Because the SIMTB1 program option to put Figures 3(a), 3(b) and 3(c) on the same scale was in effect, it may be that no boxplot in a given Figure (e.g., Figure 3(b)) requires the full range of Y-values.

In order to produce Figure 3(b), the Normal (0,1) data that went into Figure 1(a) was squared to create longer tailed $\chi^2(1)$ random variables. The output is entirely analogous to that for Figure 3(a). Similarly, for Figure 1(c), the Normal (0,1) data was exponentiated in order to create Lognormal (0,1) data and to produce analogous graphical output. The indication is that the distribution of the sample serial correlation is robust with respect to the population distribution.

The features of the SIMTB1 output will become clearer when they are associated with the various properties of the correlation estimator. First, however, a few technical comments concerning the regressions are necessary.

C. SOME COMMENTS ON THE REGRESSIONS

Two types of problems, numerical and statistical, can occur when attempting to fit the two sets of regression equations presented in Section VI.B.

First, there is the question of numerical stability when the independent variables, $\{1, n_i^{-1}, n_i^{-2}, n_i^{-3}\}$ or $\{n_i^{-1}, n_i^{-3/2}, n_i^{-2}, n_i^{-5/2}\}$ decrease geometrically. If we

attempt to form X^TX, where X is the respective design matrix and X^T is the transpose of X, we get values that range from 8 (assuming 8 subsample sizes) to $\sum_{i=1}^{8} n_i^{-6}$ for the regression on the means, and $\sum_{i=1}^{8} n_i^{-2}$ to $\sum_{i=1}^{8} n_i^{-5}$ for the regression on the variances. Experience has shown that attempts to solve systems with such extremes in the X^TX matrix produce erroneous results. Instead, SIMTB1 scales the design matrices by multiplying each column of X by $\operatorname*{Max}_i(n_i)$ raised to the proper power so that no entry becomes too small. The standard Choleski decomposition is then used to fit the equations, and the coefficients are properly rescaled before they are reported. This procedure produces numerically reliable results.

The second problem concerns the breakdown of statistical assumptions in our regression models. It has already been pointed out in Section VI.B that the two sets of dependent variables:

(1) the $\bar{\Theta}(n_i)$ when considering the regression on the means;

(2) the $s^2(n_i) = \sum_{j=1}^{M\lfloor N/n_i \rfloor} \frac{(\hat{\Theta}_j(n_i) - \bar{\Theta}(n_i))^2}{M\lfloor N/n_i \rfloor}$

when considering the regression on the variances,

are not independent over i since all are based on the same simulated data. The extent of the dependence is demonstrated by the correlation matrix in Table 1. Entries in that table show the estimated correlation between $s^2(n_i)$ and $s^2(n_j)$ for

TABLE 1

ENTRIES IN THE TABLE ARE THE ESTIMATED CORRELATIONS BETWEEN THE ESTIMATED VARIANCES OF THE r_{n_i} AT DIFFERENT SUBSAMPLE SIZES:

$$\mathrm{CORR}(s^2(r_{n_i}), s^2(r_{n_j})) \quad \text{for} \quad i = 1,\ldots,8, \quad j = 1,\ldots,8$$

j \ i	1	2	3	4	5	6	7	8
1	1.00	.49	.46	-.26	.18	-.17	.14	.01
2	.49	1.00	.40	.55	.11	.38	.38	-.03
3	.46	.40	1.00	.23	.23	.44	.21	.29
4	-.26	.55	.23	1.00	.42	.86	.57	.35
5	.18	.11	.23	.42	1.00	.71	.43	.59
6	-.17	.38	.44	.86	.71	1.00	.45	.53
7	.14	.38	.21	.57	.43	.45	1.00	.72
8	.01	-.03	.29	.35	.59	.53	.72	1.00

Recall that r_n is the estimated serial correlation for a simulated Normal (0,1) subsample of size n. Also, the estimated correlations shown above were computed using 10 values (replications) of $s^2(r_{n_i})$ and $s^2(r_{n_j})$ for each i and j.

all i and j, where the estimation was done by repeating the SIMTB1 experiment with 10 different batches of 50,000 simulated random variables. Since only 10 values went into each correlation calculation, the table is only accurate to within approximately $\pm 2/\sqrt{10} = .632$. We see some indication of positive correlation, especially when i and j are close, but the lack of independence is not severe enough to hurt the regression results for either the estimated means or variances significantly.

A second assumption, implicit in any regression, is that the dependent variables have equal variances. This condition holds true for the means, which can be shown to satisfy

$$\mathrm{Var}(\overline{\Theta}(n_i)) = \frac{M}{N}$$

independently of i. The estimated variances, however, are not equivalent and, if we assume the $\hat{\Theta}_j(n_i)$ to be approximately Normally distributed so that

$$\sum_{j=1}^{M\lfloor N/n_i \rfloor} (\hat{\Theta}_j(n_i) - \overline{\Theta}(n_i))^2$$

is approximately proportional to a $\chi^2_{M\lfloor N/n_i \rfloor - 1}$ random variable, we can compute

$$\mathrm{Var}(s^2(n_i)) \cong \frac{2}{MNn_i - n_i^2}$$

To correct this problem, SIMTB1 scales the $s^2(n_i)$ by $\sqrt{n_i}$ so that

$$\mathrm{Var}(\sqrt{n_i}\ s^2(n_i)) \ \tilde{=} \ \frac{2}{MN - n_i} \ \tilde{=} \ \frac{2}{MN}$$

since $MN >> n_i$. The design matrix is scaled accordingly and the values b_0, b_1, b_2, and b_3 discussed in Section VI.B. are reported.

Table 2 shows the effects of the rescaling by presenting first the estimated variances of the $s^2(n_i)$, where the estimation is done by repeating SIMTB1 for 10 batches of 50,000 simulated data points. These estimated variances decrease as n_i increases, closely paralleling the second line of Table 2 which has the approximate theoretical values (i.e., $2/(MNn_i - n_i^2)$). The final line of Table 2 shows the estimated variances of the rescaled $s^2(n_i)$, i.e., the $\sqrt{n_i}\ s^2(n_i)$, which, as expected and hoped, show a more constant variance with i.

Although future versions of SIMTB1 will include more sophisticated regression routines and the ability to generate independent samples at each subsample size, the SIMTB1 is quick, usable, and accurate for most situations.

D. INTERPRETING THE SERIAL CORRELATION RESULTS

Returning to Figure 3(a) which shows the simulated distribution of the serial correlation coefficient from independent, Normal (0,1) data, the following comments summarize the most striking features:

TABLE 2

A COMPARISON OF THE ESTIMATED VARIANCE OF $s^2(r_{n_i})$ WITH THE APPROXIMATE THEORETICAL VARIANCE OF $s^2(r_{n_i})$ AND WITH THE APPROXIMATELY EQUIVARIANT SCALED VERSIONS, $n_i^{-.5}\, s^2(r_{n_i})$.

All entries have been multiplied by 10^5.

$n_i =$	10	20	30	40	50	75	100	150
$\hat{Var}(s^2(r_{n_i}))$	.177	.150	.204	.079	.047	.031	.049	.022
Approx. Theoretical $Var(s^2(r_{n_i}))$	.400	.200	.133	.100	.080	.053	.040	.027
$\hat{Var}(n_i^{-.5}\, s^2(r_{n_i}))$	1.77	2.99	6.12	3.18	2.33	2.33	4.88	3.35

The estimated variances of $s^2(r_{n_i})$ and $\sqrt{n_i}\, s^2(r_{n_i})$ were calculated using 10 independent replications of $s^2(r_{n_i})$.

(a) The boxplots appear very symmetric at all subsample sizes with nearly equal numbers of outliers at either tail and with mean and median coincidental. This observation is confirmed by the estimates of skewness in the summary statistics. Kurtosis is mildly negative at small subsample sizes but, overall, asymptotic Normality seems to take hold rather quickly.

(b) The average serial correlation is negative for small subsam-les. This is demonstrated by the dotted regression curve which starts at approximately -.10 and levels off near 0 for subsamples greater than about 85. The dashed asymptote of -.000272 is very close to the theoretical value of 0, and the mean values in the summary table closely reflect the bias of -1/n.

(c) The standard deviations of the simulated distributions are very close to the asymptotic values of $n_i^{-0.5}$, although the lead coefficient in the regression on the variances, b_0 = .743756, is not as close to the theoretical value of 1 as we would hope. When SIMTB1 is repeated 10 times with 10 different batches of simualted data, we find an average value for b_0 to be 1.0604, with a standard deviation for b_0 of .307. The estimation procedure for b_0, therefore, remains valid, but the estimate itself is highly variable.

The agreement between the simulated and the theoretical, asymptotic values of the bias and variances was discovered previously by Cox (1966). SIMTB1 has now allowed us to

automatically look at a broader range of subsample sizes and to see, through boxplots and estima es of skewness and kurtosis, a fuller picture of any changes in the distribution of the estimator. We can be satisfied that estimates of serial correlation do behave approximately as Normal (-1/n, 1/n) rangome variables when the underlying data is Normal (0,1).

If the lead terms in the expansions of the mean and variance of the estimated correlation coefficient (i.e., a_0, a_1, and b_0) had been unknown, we would also have a fairly good idea now of what they were.

When the underlying data is χ_1^2, Figure 3(b) confirms Cox's observation that the bias is relatively uneffected but, for small subsamples, the standard deviation is smaller than the expected $n^{-1/2}$. Unlike Figure 3(a), there is a pronounced skewness in the boxplots in Figure 3(b) with many more outliers at the positive end, and with the mean higher than the median at the first four subsample sizes. The problem of suppressed variance seems cured at $n_7 = 100$ and $n_8 = 150$, but the skewness remains and could cause problems in tests of significance.

Figure 3(c), which is based on an underlying batch of simulated Lognormal (0,1) data, shows a slight exaggeration of the effects in Figure 3(b). The standard deviation is more suppressed and does not attain the theoretical level by $n_8 = 150$. The positive skewness is more pronounced and kurtosis does not approach the theoretical value of 0.

Overall, the effects of long-tailed data on the distribution of the serial correlation coefficient can be summarized as follows:

(i) Bias is not significantly effected and remains at approximately $-1/n$.

(ii) The variance of the distribution of the serial correlation coefficient is reduced by longer-tailed data.

(iii) Positive skewness is created in the distribution.

(iv) Kurtosis may become positive at larger subsample sizes.

(v) For long-tailed data (i.e., Lognormal), a subsample size of 150 is not large enough to insure asymptotic Normality.

E. SIMTB1 OUTPUT FOR THE Z-TRANSFORM OF THE CORRELATION

Fisher's z-transform of the estimated correlated coefficient is defined by:

$$Z_n = \frac{1}{2} \log \frac{1 + r_n}{1 - r_n} ,$$

where r_n is the estimated serial correlation presented in Section VI.B. The transformation is intended to make the distribution of the Z_n more Normal than that of the r_n. When the same SIMTB1 experiment described in Section VI.B. is run using Z_n as the estimator instead of r_n, we get the results shown in Figures 4(a), 4(b) and 4(c). It should

be noted that the scale of the boxplots here has been forced to be approximately comparable to the scale for the boxplots in Section VI.B. This is done by suppressing outliers that are more than 1.5 interquartile distances beyond the quartiles of the boxplot. If we had allowed the data to scale the boxplots, we would have seen a much wider range on the vertical axis because the Z_n are not restricted to the limits of -1 to +1 and because there is one far outlier at -3.8. In this type of "reduced graphics," we still see the number of outliers that fall beyond the allowable range through the numbers at the ends of the boxplots, but we do not see their actual locations.

Figure 4(a) shows the distribution of the z-transformed correlation coefficients when the underlying data is simulated, Normal (0,1). At each subsample size, the mean and standard deviation are close to the theoretical n^{-1} and $n^{-1/2}$ respectively. The skewness and kurtosis at subsample size $n_1 = 10$ are far from the theoretical Normal distribution values of 0 and 0, reflecting partly the one far outlier at -3.8 and partly the negative skew in the remainder of the Z_{n_1}'s. For other subsample sizes, there is no strong evidence to contradict the assumption of approximate Normality.

The relationship between Figure 4(b) and 4(a) is similar to that between 3(b) and 3(a). Figure 4(b), which is based on simulated χ_1^2 data, shows (a) bias that is the same as for the transformed correlations based on Normal data, (b) slightly

suppressed variances, particularly at small subsample sizes and (c) positive skewness which persists at large subsample sizes. In addition, there are signs of positive kurtosis at small subsample sizes.

Figure 4(c) is based on Lognormal (0,1) data and shows high values of skewness and kurtosis at almost all subsample sizes. Approximate Normality seems an unwarranted assumption. In fact, the kurtosis is converging very slowly to its asymptotic value of 0.

In general, using the z-transform does not help with Normality assumptions, especially when dealing with long-tailed distributions.

F. SIMTB1 OUTPUT FOR THE 2-FOLD JACKKNIFE OF THE CORRELATION

The final Figures, 5(a), 5(b) and 5(c), deal with the 2-fold jackknife estimate of correlation. Again, the figures are reduced graphics with scaling comparable to that of the boxplots of Sections VI.D and VI.E. To define the estimator, we start with a given subsample of size n, compute the serial correlation for the first $\lfloor n/2 \rfloor$ points and call it $r_1(n/2)$, compute the serial correlation for the second $\lfloor n/2 \rfloor$ points and call it $r_2(n/2)$ and compute the serial correlation for the entire subsample of n points and call it $r_0(n)$. Each computation follows the formula in Section VI.B. The three estimators are then combined to form two pseudo-values,

$$r_1^*(n) = 2r_0(n) - r_1(n/2)$$

and

$$r_2^*(n) \;=\; 2r_0(n) - r_2(n/2) \;,$$

and the final jackknife estimator for that subsample is defined as

$$\tilde{r}(n) \;=\; \frac{r_1^*n) + r_2^*(n)}{2} \;.$$

Although a jackknife estimator may have many favorable properties, we are concerned here primarily with its ability to remove bias, hopefully without inflating the variances of the estimator and/or inducing nonnormality.

Figure 3(a), based on simulated Normal (0,1) data, shows nearly complete removal of bias, even at small subsample sizes. The cost of the bias reduction is reflected in an increase of nearly 50% in the standard deviation of the correlation estimate for subsample size 10, and lesser relative increases at larger subsample sizes. There is also an indication of a positive skew for small subsample sizes, and the problem that the jackknife estimator need not fall into the -1 to +1 range which is desirable for a correlation coefficient estimate.

When using simulated χ_1^2 data as in Figure 5(b), or simulated Lognormal (0,1) data as in Figure 5(c), there is again no problem with bias. Variance inflation, though it exists

at small subsample sizes, is not as large as when Normal (0,1) data is used. The distributions of the jackknifed correlations show very pronounced positive skews, however, as well as positive kurtosis. These two problems are worse for the longer-tailed Lognormal data.

Overall, the jackknife estimator is very successful at removing bias but the costs include variance inflation, which can be severe at small subsample sizes, plus increased positive skewness and kurtosis when the estimates are based on data from longer-tailed distributions.

G. COMPARISON OF THE THREE ESTIMATES OF CORRELATION

For Normal (0,1) data, the distribution of the usual correlation coefficient displayed in Figure 3(a) behaves very much as theoretical asymptotic calculations would predict, even at small subsample sizes. This makes it possible to correct for bias in the estimator and to perform tests of significance. Use of Fisher's z-transform, as illustrated in Figure 4(a) does not seem necessary since it does not significantly improve the approximate Normality of the estimator. The jackknife estimator in Figure 5(a) may be valuable if a direct, unbiased estimator is needed but the inflated variance of the jackknife estimator may limit the usefulness of the estimate as well as make any tests of significance too conservative.

When the underlying data comes from a longer-tailed distribution, the usual correlation coefficient in Figures 3(b)

and 3(c) retains a predictable bias term, although the variance of its distribution is slightly depressed and the skewness and kurtosis becomes positive, even for subsamples as large as 150. This means that it is still possible to estimate the correlation accurately, but tests of significance fall on shakey assumptions of Normality. The z-transform in Figures 4(b) and 4(c) does little to firm up those assumptions and, in some cases, makes the situation worse. As in the case of Normal data, the 2-fold jackknifed correlation in Figures 5(b) and 5(c) is bias-free but follows a fairly non-Normal distribution which would invalidate significance testing.

All of the preceding observations and conclusions flow immediately from the nine figures presented so far. Further studies could easily be done through SIMTB1, looking at larger subsample sizes, correlated data, and alternative marginal distributions. For demonstration purposes, though, it is better to proceed to our second application.

VII. STUDY OF PROBLEMS OF ESTIMATING SHAPE PARAMETERS FOR HIGHLY SKEWED DISTRIBUTIONS

A. ESTIMATING THE SHAPE PARAMETER FOR A GAMMA DISTRIBUTION

As a second application of SIMTB1, we will consider a problem which has received much less statistical attention; asymptotic results are summarized in Cox and Lewis (1966, Ch. 2) [Ref. 7] and Johnson and Kotz (1970, Ch. 17) [Ref. 8]. We want to estimate the shape parameter, K, for a Gamma distribution, where the Gamma density is given by

$$f(x) = \begin{cases} (\frac{K}{u})^K \frac{x^{K-1}}{\Gamma(x)} e^{-Kx/u} & x > 0; \quad K > 0; \quad \mu > 0 \\ 0 & x < 0 \end{cases}$$

Notice that the mean of this distribution is u, not K/u as in some differently parameterized versions of the Gamma density. For the data that will be simulated for use in SIMTB1 we will use K = 5 and u = 1 and K = 0.25 and μ = 1. The closer the mean of our estimate is to 5 or 0.25, the better (in terms of bias) is our estimation procedure. Other factors such as the variance and Normality of the estimator will of course also have influence in the determination of a preferred estimator.

Section VII.B will compare the commonly used maximum likelihood estimator to the competing method of moments

estimator. Both procedures result in asymptotically Normal estimators (Cramer, 1948) but the m.l.e. is usually preferred because of its favorable asymptotic relative efficiency (Cox and Lewis, 1966) [Ref. 7]. Through SIMTB1, though, we will see that for small subsamples the estimated variances of the two estimators of K are not as far apart as asymptotic results lead us to believe. In addition, the bias that appears in both estimators is smaller for the moment estimator.

In Section VII.C. we will use a four-fold jackknife of both the m.l.e. and moments estimators to successfully remove the bias. What is remarkable is that, unlike the jackknifing of the serial correlation, there is little or no cost in terms of variance inflation and nonnormality for the jackknifed moment estimator. When K = .25, we will see in Section VII.D. that the jackknifed m.l.e. dominates the other three estimators at all subsample sizes when considering the mean, variance, and Normality of the estimator.

B. MAXIMUM LIKELIHOOD AND MOMENT ESTIMATORS OF K

Figure 6(a) is very similar in format to the figures that have already been presented for the correlation example <u>except</u> that:

(1) The estimator whose distribution is being displayed is the maximum likelihood estimator of K, the shape parameter of a Gamma(5) population. We denote the estimator, computed from a simulated subsample of size n, by $\hat{K}(n)$ and define it

to be the solution of the equation:

$$n[\log \hat{K}(n) - \Psi(\hat{K}(n))] = n \log \sum_{i=1}^{n} X_i/n - \sum_{i=1}^{n} \log x_i ,$$

where the X_i are the simulated Gamma(5) random variables and $\Psi(\cdot)$ is the digamma function (Cox and Lewis, 1966).

(2) The eight subsample sizes which we will be looking at are $n_1 = 33$, $n_2 = 50$, $n_3 = 71$, $n_4 = 100$, $n_5 = 125$, $n_6 = 166$, $n_7 = 250$ and $n_8 = 500$. We will not see as much detail at small subsample sizes but we will see some of the asymptotic ($n = 500$) effects coming in.

(3) At each subsample size we will work with $M^* = 20$ independent replications of $N^* = 2500$ simulated Gamma(5) random variables, instead of the $M = 10$ replications of $N = 5000$ variables used previously. The total number of independent simulated random variables across replications remain constant at the program maximum of 50,000. Hence, the boxplot at subsample size 50 in Figure 6(a) represents the distribution of $M^* \lfloor N^*/50 \rfloor = 1000$ estimates of $\hat{K}(50)$ just as the boxplot at subsample size 50 in Figure 3(a) represents the distribution of $M \lfloor N/50 \rfloor = 1000$ estimates of $r(n)$. As long as the product, $M \times N$, remains constant, the only effect that changing the number of replications has, up to rounding in $\lfloor N/n_i \rfloor$, is to change the results in the regression on the averages. By using $M^* = 20$ and $N^* = 2500$, SIMTB1 reports regression coefficients averaged over 20 replications, but,

within each replication, the dependent variables are averages over just $\lfloor 2500/n_i \rfloor$ values of the estimator.

(4) The boxplots are presented using the reduced graphics option. In this option any extreme outliers (i.e., those beyond 1.5 interquartile distances) are included as a count at the tail of each boxplot. This option was chosen in order to give more graphical weight to the body of the distributions and the fall-off in the bias. Limited printer resolution makes it impossible to show details in the body and the tails of the distributions if there are many straggling outliers. In the case of very extreme outliers, no detail would be seen in the body of the boxplot without the reduced graphics option.

Figure 6(b) looks at the distribution of the moment estimator of K, the shape parameter of a Gamma (K) population:

$$\tilde{K}(n) = (n-1)\,\bar{X}^2 / \sum_{i=1}^{n} (X_i - \bar{X})^2 ,$$

where $\bar{X} = \sum_{i=1}^{n} X_i/n$, n is the subsample size, and the X_i are the simulated Gamma(5) random variables. The SIMTB1 options and parameters mentioned in (2), (3) and (4) preceding are also in effect here.

The two Figures, 6(a) and 6(b), show a very pronounced bias in both estimation procedures, although the moment estimator is slightly closer to the unbiased value of 5. As expected, the standard deviation of the m.l.e. is lower than that of the moment estimator although the relative

difference at small subsample sizes, for instance 1.448 versus 1.482 at $n_1 = 33$, may not outweight the increase in bias with the m.l.e. At larger subsample sizes, the relative difference is close to the theoretical asymptotic relative efficiency of .78 (i.e., .91 at $n_7 = 250$).

Both estimators also show distributions with positive skewness and kurtosis that decrease to the asymptotic 0 levels as subsample size increases. The asymptotics appear to take hold more quickly for the moment estimator than for the m.l.e.

In summary, SIMTB1 shows that the m.l.e. is indeed better than the moment estimator in terms of variance, but not as good for small sample sizes as asymptotic results would lead us to believe. In the other areas of bias and asymptotic Normality, the moment estimator would have to be preferred.

C. 4-FOLD JACKKNIFED ESTIMATORS OF K

Figures 6(c) and 6(d) show the distributions of the 4-fold jackknife m.l.e. of K and 4-fold jackknifed moment estimator of K, respectively. A 4-fold jackknife estimator is similar to the 2-fold jackknife estimator described in Section VI.F. except that there are 4 pseudo-values that come out of dividing each subsample into fourths. More details can be found in Mosteller and Tukey (1977) [Ref. 6].

The purpose of the jackknife is to remove the conspicuous bias observed in Figures 6(a) and 6(b). This goal is seen to be accomplished in Figure 6(c) and 6(d) and we can also note

smaller values of skewness and kurtosis, indicating a quicker approach to asymptotic Normality. The skewness and kurtosis of the jackknifed moment estimator are the lowest, at small subsample sizes, among all estimators. The variance of the jackknifed moment estimator is also only slightly inflated, as is the variance of the jackknifed m.l.e.

All told, the jackknifed moment estimator, because of its lack of bias, small variance, and low skewness and kurtosis, would be the method of choice if estimation of K or significance testing was the goal.

D. RESULTS FOR K = 0.25

In Figures 7(a), 7(b), 7(c) and 7(d) we show similar results to those discussed above for the case K = 5.0, but using K = 0.25. The fact (Cox and Lewis, 1966, Ch. 3) [Ref. 7] that the m.l.e. estimate is much more efficient than the moment estimate is graphically illustrated. What is new is the effect of jackknifing: bias is reduced without the sacrifice of variance inflation or nonnormality.

Further comparisons and interpretations are similar to those done for the case K = 5.0, and are left to the reader.

E. CONCLUSIONS

Simply by providing SIMTB1 with the desired estimators, we have been able (a) to explore in depth the effects of changes in data distribution and of different estimation procedures on the calculation of the serial correlation

coefficient, and (b) to compare four different ways to estimate the shape parameter in a highly skewed Gamma population.

The graphics and numerical output combine to let us see and quantify distributional changes that occur as subsample size grows. We can see bias fall away, variance shrink, and skewness disappear as the estimator approaches asymptotic Normality. Terms in the asymptotic expansion of the mean and variance of the estimator are automatically calculated and can be used to compare different estimators.

VIII. COMPARISON OF DIFFERENT METHODS FOR ESTIMATING THE VARIABILITY OF THE STANDARD DEVIATION OF CORRELATION ESTIMATES

A. INTRODUCTION

Bradley Efron and Gail Gong (1982) review in their article, "A Leisurely Look at the Bootstrap, the Jackknife, and Cross-Validation" [Ref. 9] different methods for estimating statistical error of parameter estimates. They discuss in particular the problem of estimating the error of a statistical estimator with the example of the estimates of the standard error for the correlation coefficient of a bivariate normal distribution. They compared the standard deviation estimates of different methods using 200 simulations at one fixed sample size of 14.

SIMTB2 was used to explore the distributions of the estimates they used in their article. The bootstrap, the jackknife, the infinitesimal jackknife (delta method) and the normal theory are the methods for which the distributions were explored. The estimation methods will not be explained in detail (see Efron and Gong in The American Statistician, Feb. 1983, Vol. 37, No. 1), only the setup of SIMTB2 and the output will be discussed.

B. SETUP OF SIMTB2 AND THE ESTIMATOR FUNCTIONS

The bivariate normal distributed input data with known correlation 0.5 was generated with an IMSL random number

generator (GGNSM). 14,000 bivariate data points were generated and then sectioned into 10 blocks, each having 1,400 points (M = 10, N = 1400). The subsample sizes (NE(k)) used are 10, 14, 20, 28, 35, 40, 70 and 100. Only subsample size 14 was used by Efron and Gong.

The program had to be run with 5 different standard deviation estimator functions. SIMTB2, as the other versions, can only handle up to 3 estimator functions in one program run. To make the given outputs comparable the fixed scale option (SVS = 1, YMIN = 0.0 and YMAX = 0.7) was chosen.

The bootstrap estimate of standard deviation was done with 2 different numbers of bootstrap replications (B = 128, B = 512). So for the subsample size of 14, the bootstrap procedure with B bootstrap replications in itself was done (1400/14 = 100; 100*10 = 1000) 1,000 times.

The jackknife function followed the standard jackknife procedure and used the jackknife formula for the standard deviation:

$$S_{jack} = \left[\frac{n-1}{n} \sum_{i=1}^{n} (\overline{X}_{(i)} - \overline{X}_{(\cdot)})^2\right]^{1/2}$$

with

$$\hat{\Theta}_{(j)} = \hat{\Theta}(x_1, x_2, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) \quad \text{for} \quad \overline{X}_{(i)}$$

and

$$\hat{\Theta}_{(\cdot)} = \frac{1}{n} \sum_{i=1}^{n} \hat{\Theta}_{(i)} \quad \text{for} \quad \overline{X}_{(\cdot)}$$

The delta method (infinitesimal jackknife) function calculated the estimate with the formula:

$$S_{Delta} = \left\{ \frac{\hat{\rho}^2}{4n} \left[\frac{\hat{\mu}_{40}}{\hat{\mu}_{20}^2} + \frac{\hat{\mu}_{04}}{\hat{\mu}_{02}^2} + \frac{2\hat{\mu}_{22}}{\hat{\mu}_{20}\hat{\mu}_{02}} + \frac{4\hat{\mu}_{22}}{\hat{\mu}_{11}^2} - \frac{4\hat{\mu}_{31}}{\hat{\mu}_{11}\hat{\mu}_{02}} - \frac{4\hat{\mu}_{13}}{\hat{\mu}_{11}\hat{\mu}_{02}} \right] \right\}^{1/2}$$

with

$$\hat{\mu}_{gh} = \frac{1}{n} \sum_{i=1}^{n} (Y_i - \overline{Y})^g (Z_i - \overline{Z})^h$$

and

$$X_i = (Y_i, Z_i)$$

In doing the actual calculations for the delta method, for small subsample sizes (10,14,...,40) the value of the variance becomes negative. A negative variance can not be interpreted with much meaning. The number of negative values goes down (from 305 for NE(1) = 10 to 18 for NE(6) = 40) with increasing subsample size. Most of the negative values are small. To solve the programming problem (square root of a negative number) the function sets this negative value to 0.0D0.

For the normal theory estimate, instead of carrying out a bootstrap, an approximation formula was used. It is the

same formula the article uses for the comparison.

$$S_{norm} = \frac{(1 - \hat{\rho}^2)}{\sqrt{(n - 3)}}$$

See Efron and Gong for details.

C. INTERPRETING THE SIMTB2 OUTPUT

The output of the program runs is provided as Figures 6, 7, 8, 9 and 10. A comparison of the numerical output (subsample size 14) with the results is done in Table 3.

The bootstrap procedure (Figures 8 and 9) was done with 2 different numbers of bootstrap replications (B = 128 and B = 512). Both distributions for the standard deviation (S.D.) look very similar. Both are positively skewed with some outliers at the right tail. In both cases the outliers are in the same range. As the boxplots and the summary statistics show, the increase of the number of bootstrap replications (B) does not result in a large improvement in the performance of the estimation function.

The jackknife estimate (Figure 10) has a positively skewed distribution with outliers. The distribution of the jackknife estimate looks very similar to the distribution of the bootstrap estimates. For small subsample sizes the bootstrap distribution has more outliers. Overall the performance of the jackknife procedure is as good as the bootstrap, but the jackknife needs less computer time.

TABLE 3

ESTIMATES OF THE STANDARD DEVIATION FOR THE CORRELATION COEFFICIENT FOR A BIVARIATE NORMAL WITH TRUE CORRELATION $\rho = .5$

	Summary Statistic 200 Trials (Efron & Gong (1982))				Summary Statistic 1000 Trials SIMTB2 (M = 10, N = 1400, NE(k) = 14)			
	Ave	S.D.	CV	$\sqrt{MSE}$	Ave	S.D.	CV	$\sqrt{MSE}$
Bootstrap B = 128	.206	.066	.32	0.062	.212	0.059	.28	0.059
Bootstrap B = 512	.206	.063	.31	0.064	.212	0.058	.27	0.058
Jackknife	.223	.085	.38	0.085	.226	0.086	.38	0.086
Delta Method	.125	.058	.33	0.022	.157*	0.096*	.61*	0.074*
Normal Theory	.217	.056	.26	0.056	.217*	0.055	.25	0.055
True Value	.218							

*negative values set to 0.0

The distribution of the estimates produced by the delta method (Figure 11) is negatively skewed and has nearly no outliers. But in calculating the estimates the problem of negative values for the variance came up. For some subsamples, the final estimate (the standard deviation) could not be calculated, since the corresponding value of the variance was negative. In these cases, the standard deviation was set to 0.0. This procedure influences the distribution and the summary statistics. The influence is more important for small subsample sizes than for larger ones. So the graphical and numerical output should be seen with this fact in mind.

The normal theory function (Figure 12) produces estimates with a negatively skewed distribution but only a few outliers and the distribution is tailed to the left. The tail of the distribution is in the opposite direction of all other distributions. For the estimate of the standard deviation for the correlation coefficient the result of the normal theory is close to the true result. This may not be valid for other estimators.

In addition to the comparisons Efron and Gong did, with SIMTB2 it is easy to investigate how the sample size will influence the estimate of the standard deviation of the correlation coefficient. In Table 4 the methods are compared for a subsample size of 10 and 100. With increasing subsample size the quality of the estimate should improve, but the

TABLE 4

ESTIMATOR OF THE STANDARD DEVIATION FOR THE CORRELATION COEFFICIENT FROM A BIVARIATE NORMAL WITH TRUE CORRELATION ρ = 0.5 AT DIFFERENT SAMPLE SIZES. SIMTB2(M = 10, N = 1400)

	1400 Trials Subsample Size 10				140 Trials Subsample Size 100			
	AVE	S.D.	C.V.	$\sqrt{MSE}$	AVE	S.D.	C.V.	$\sqrt{MSE}$
Bootstrap B = 128	0.26	0.083	0.32	0.083	0.087	0.0076	0.088	0.013
Bootstrap B = 512	0.26	0.082	0.32	0.082	0.087	0.0077	0.088	0.014
Jackknife	0.28	0.13	0.46	0.13	0.076	0.011	0.14	0.011
Delta Method	0.18*	0.12*	0.64*	0.148*	0.071*	0.016*	0.22*	0.016*
Normal Theory	0.027	0.082	0.31	0.082	0.076	0.0076	0.1	0.008
True Value	0.267				0.0753			

*negative values for Variance set to 0.0

improvement may be different for the different methods of estimation. By making the subsample size 10 times larger, with the SIMTB2-side effect of reducing the total number of calculated estimates, the bootstrap improves less than the jackknife, delta method and normal theory.

IX. FINAL CONCLUSIONS

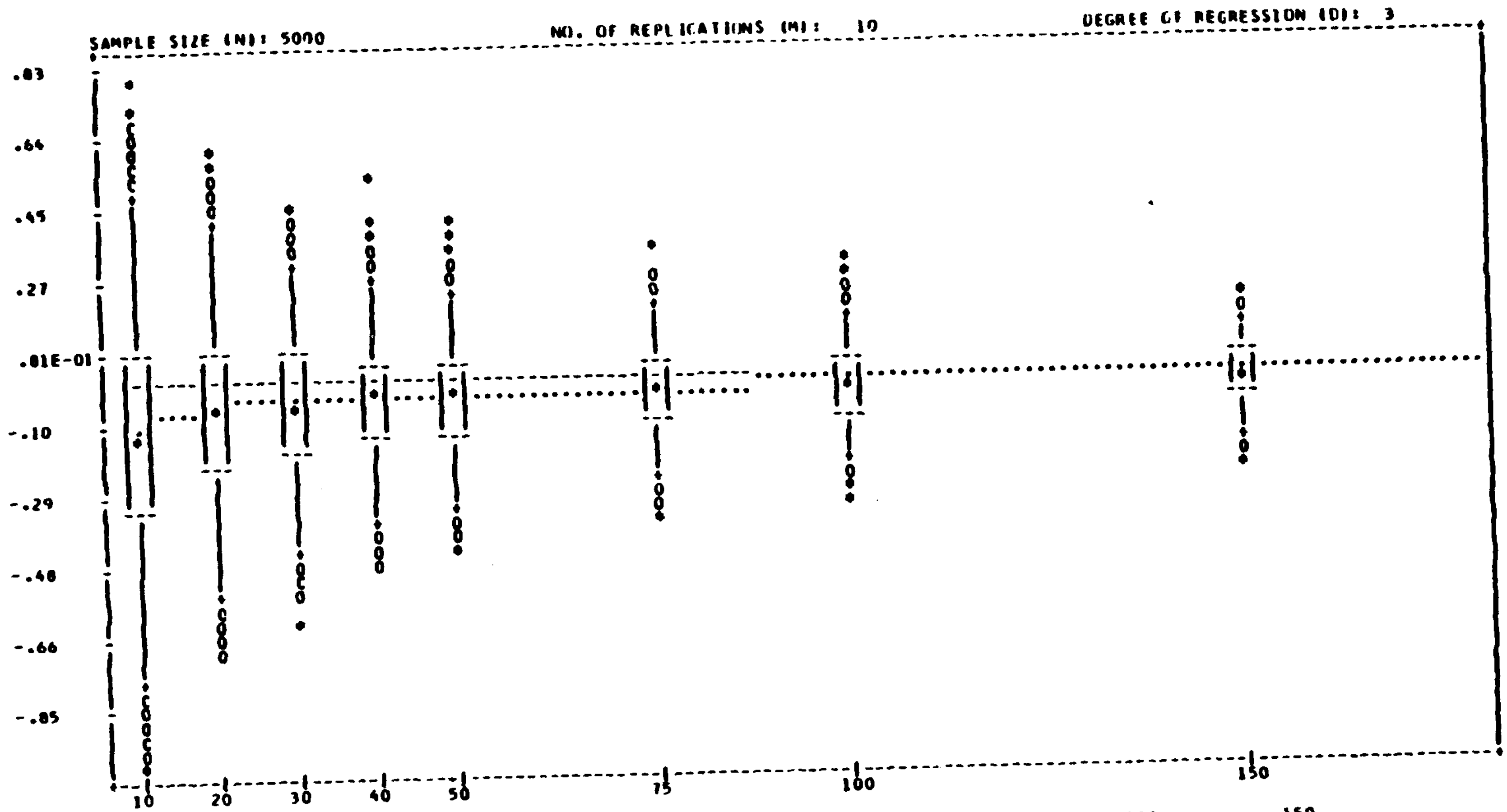
SIMTBED with the different versions can be used on digital computers of different size (mainframe to micro) and type. The limitations in using the program are given more by hardware constraints like memory size and computer time than by the program itself.

The FORTRAN program is completely portable, changes in the code, may only be necessary to adapt the program to special restrictions given by a special type of hardware. This may occur in using micro computers more often than with mainframe computers. Up to now all versions of code are written for the more normal standard computer environment and do not need special equipment (color printer, etc.). Additionally hardware dependent features like color output can improve the graphics of the program.

SIMTBED makes it easy to evaluate the result of statistical experiments. The combination of graphics and numerical summaries for different sample sizes make it easy to judge the distributional behavior of a statistical estimator. The result can be seen without additional computations in the graphic outputs. Comparing only the boxplots it is possible to judge the influence of subsample size on the variability of an estimator.

Besides for research the program can be used in showing students the distributional behavior of different estimators in a pictorial way. It is easy to compare the different behavior of similar estimators (e.g., biased vs. unbiased estimator of the variance) for different sample sizes.

The easy use and the fast visual impression of the distributional behavior of an estimator, given by the graphic output is one of the advantages in using SIMTBED. Besides this fast first visual impression all necessary and needed numerics are given for further and deeper investigations.



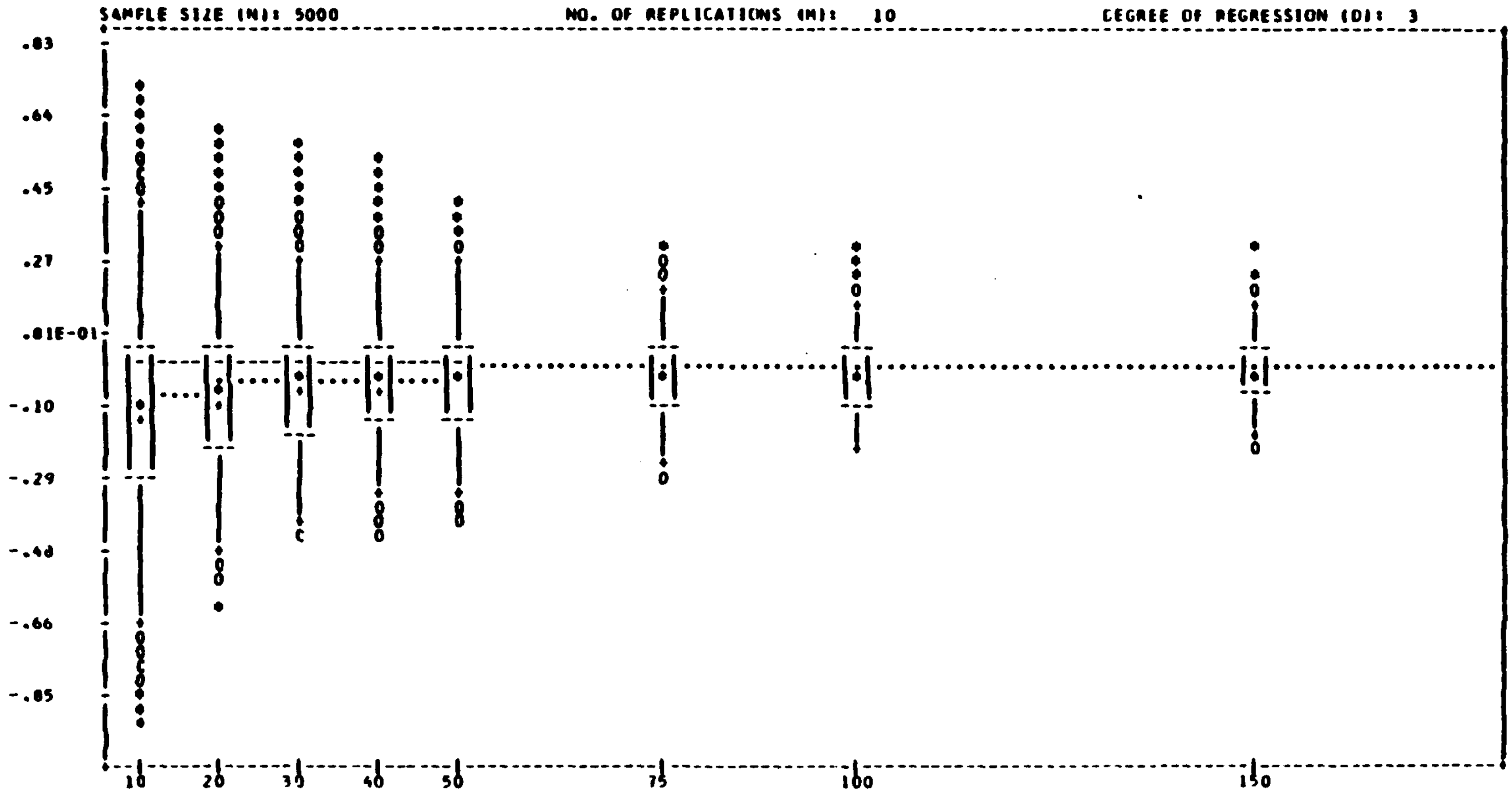
SUBSAMPLE SIZE	10	20	30	40	50	75	100	150
MEAN	-.1074	-.5125E-01	-.3465E-01	-.2611E-01	-.1980E-01	-.1442E-01	-.1008E-01	-.7372E-02
STD	.2996	.2206	.1829	.1558	.1389	.1142	.9653E-01	.7822E-01
STD MEAN	.4237E-02	.4416E-02	.4488E-02	.4405E-02	.4392E-02	.4444E-02	.4317E-02	.4306E-02
SKEWNESS	0.108	0.027	0.013	0.058	0.069	0.055	-0.066	-0.148
KURTOSIS	-0.426	-0.375	-0.274	-0.164	-0.147	-0.228	0.340	0.086

MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:	-.271774E-03	-1.03074	.694488	-11.1320
VARIANCE OF REGRESSION - COEFFICIENTS:	.151458E-04	.152014E-01	1.34950	349.482
STD DEV OF REGRESSION - COEFFICIENTS:	.389176E-02	.123294	2.11100	18.6944
REGRESSION ON VARIANCE - COEFFICIENTS:	.743756	2.85748	-9.48545	6.28721

VERTICAL SCALE: YMIN = -0.9586
YMAX = 0.8258

ESTIMATOR: ESTIMATES OF THE LAG ONE SERIAL CORRELATION COEFFICIENT FOR A NORMAL(0,1) SAMPLE

Figure 3a. Estimates of the Lag One Serial Correlation Coefficient for a Normal (0,1) Sample



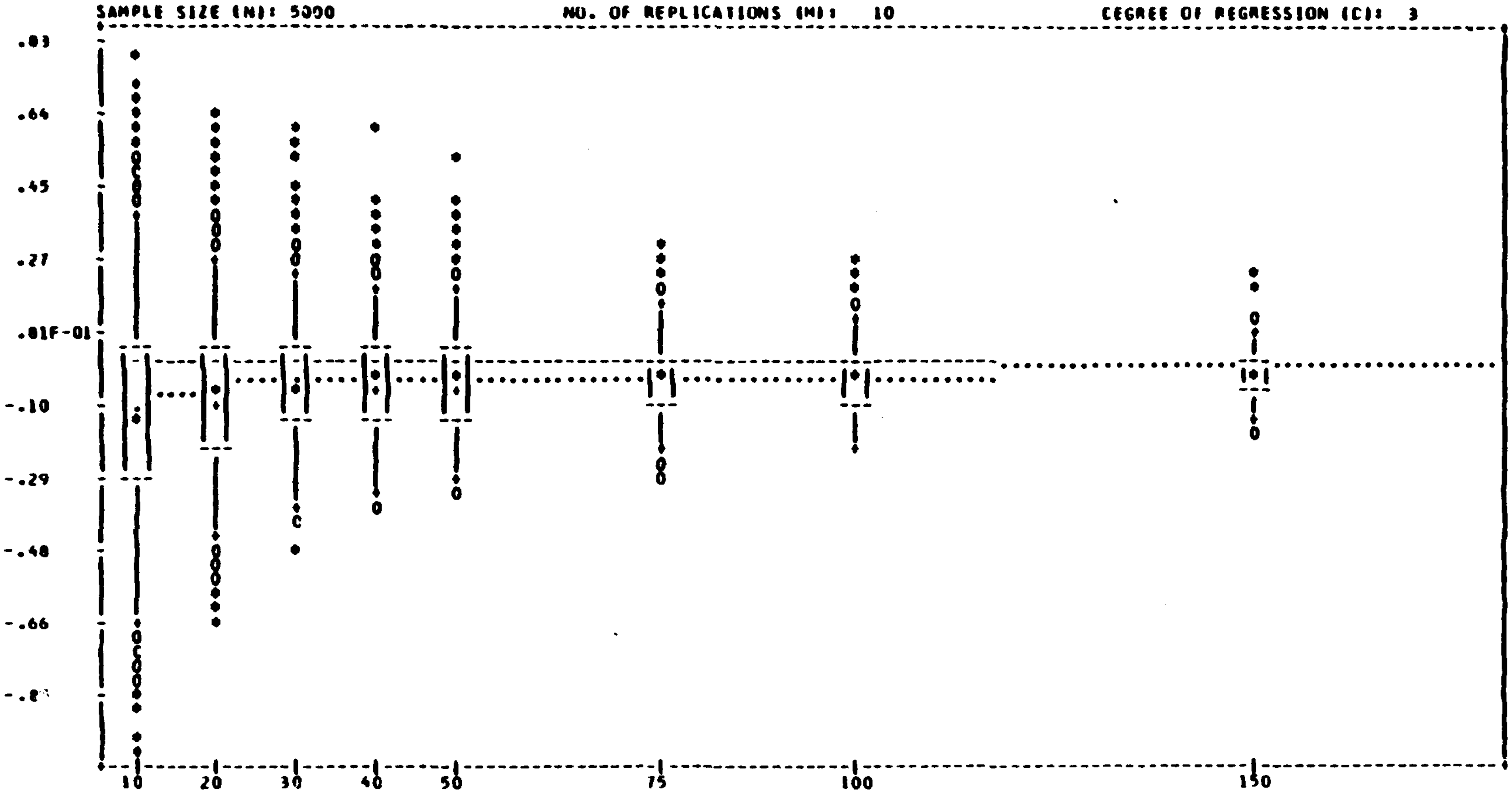
SUBSAMPLE SIZE	10	20	30	40	50	75	100	150
MEAN	-.1034	-.4659E-01	-.2814E-01	-.1817E-01	-.1248E-01	-.6363E-02	-.2613E-02	.1486E-02
STD	.2608	.1962	.1663	.1480	.1337	.1102	.9568E-01	.8356E-01
STD MEAN	.3689E-02	.3924E-02	.4080E-02	.4187E-02	.4227E-02	.4288E-02	.4275E-02	.4600E-02
SKEWNESS	0.327	0.531	0.577	0.494	0.470	0.423	0.487	0.372
KURTOSIS	-0.207	0.039	0.181	0.127	-0.049	-0.284	0.168	0.224

MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:	.775021E-02	-.974092	-3.49581	21.0796
VARIANCE OF REGRESSION - COEFFICIENTS:	.230496E-04	.140388E-01	10.0476	575.014
STD DEV OF REGRESSION - COEFFICIENTS:	.480361E-02	.118485	3.16979	23.9794
REGRESSION ON VARIANCE - COEFFICIENTS:	1.17537	-2.52788	3.87640	-2.64231

VERTICAL SCALE: VMIN = -0.9986
VMAX = 0.8258

ESTIMATOR: ESTIMATES OF THE LAG ONE SERIAL CORRELATION COEFFICIENT FOR A CHI-SQUARE(1) SAMPLE

Figure 3b. Estimates of the Lag One Serial Correlation Coefficient for a Chi-Square (1) Sample



SUBSAMPLE SIZE	10	20	30	40	50	75	100	150
MEAN	-.1098	-.5318E-01	-.3701E-01	-.2851E-01	-.2317E-01	-.1698E-01	-.1271E-01	-.1049E-01
STD	.2597	.1890	.1574	.1334	.1195	.9931E-01	.8528E-01	.6686E-01
STD MEAN	.3672E-02	.3781E-02	.3864E-02	.3773E-02	.3778E-02	.3866E-02	.3814E-02	.3681E-02
SKEWNESS	0.340	0.462	0.519	0.545	0.581	0.580	0.557	0.668
KURTOSIS	-0.027	0.243	0.280	0.434	0.682	0.507	0.725	0.901

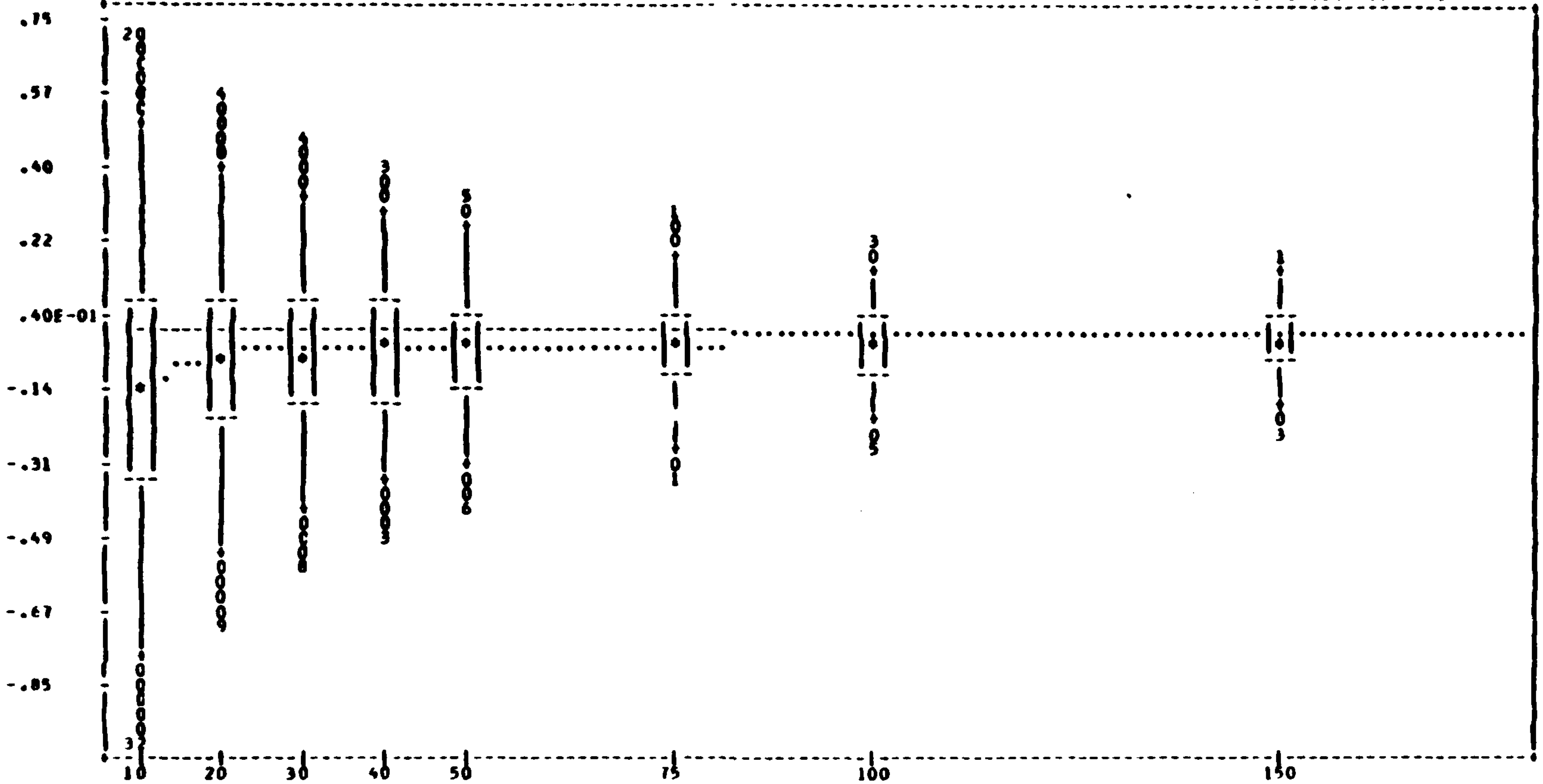
MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:	-.310053E-02	-1.04703	1.87581	-20.8023
VARIANCE OF REGRESSION - COEFFICIENTS:	.601527E-05	.435615E-01	22.9494	960.390
STD DEV OF REGRESSION - COEFFICIENTS:	.245260E-02	.208714	4.79055	30.9902
REGRESSION ON VARIANCE - COEFFICIENTS:	.563485	2.35455	-10.1745	12.1317

VERTICAL SCALE: YMIN = -0.9586
YMAX = 0.8258

ESTIMATOR: ESTIMATES OF THE LAG ONE SERIAL CORRELATION COEFFICIENT FOR A LOGNORMAL(0,1) SAMPLE

Figure 3c. Estimates of the Lag One Serial Correlation Coefficient for a Lognormal (0,1) Sample

SAMPLE SIZE (N): 5000 NO. OF REPLICATIONS (M): 10 DEGREE OF REGRESSION (D): 3



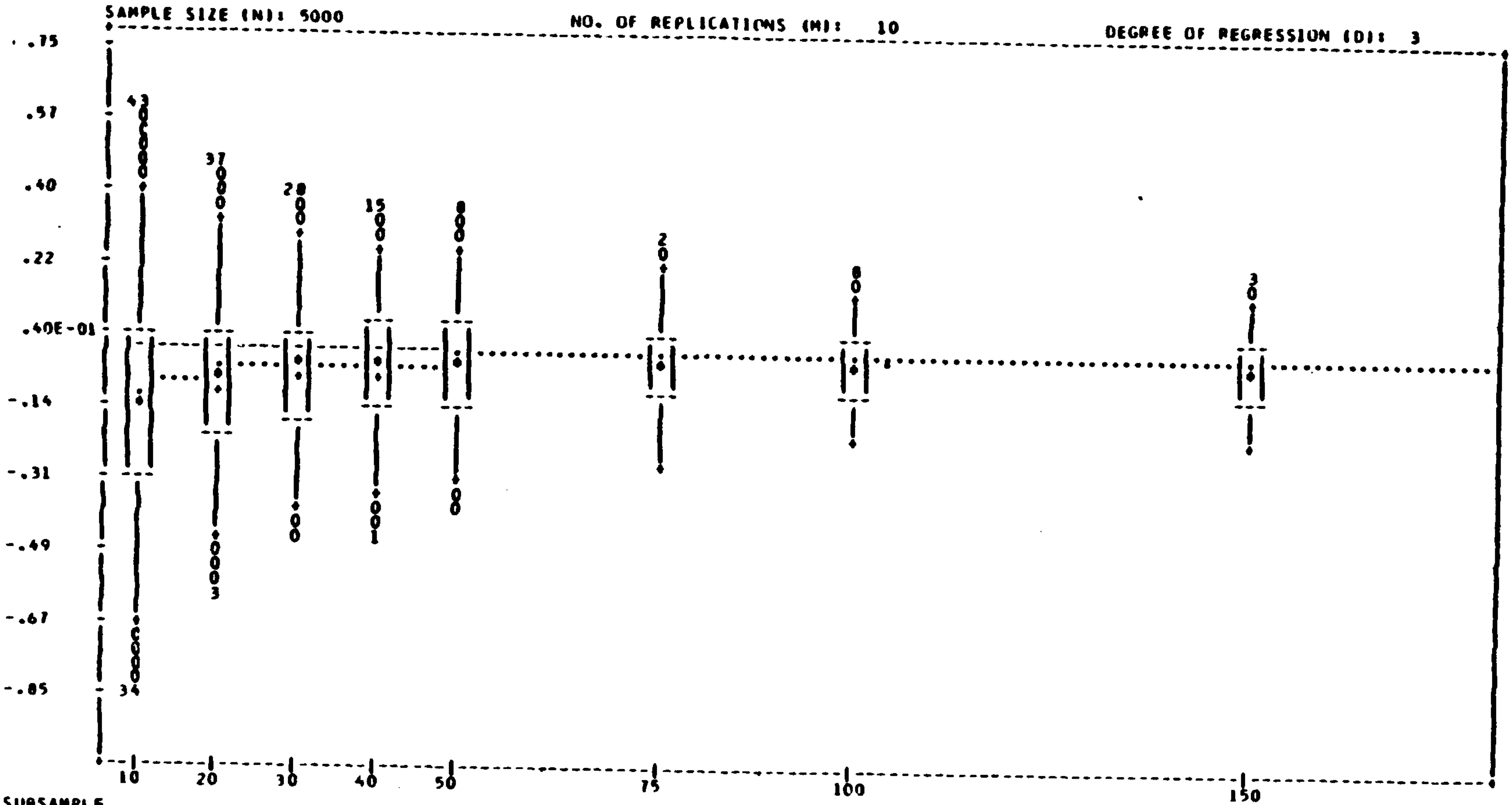
SUBSAMPLE SIZE	10	20	30	40	50	75	100	150
MEAN	-.1203	-.5408E-01	-.3592E-01	-.2671E-01	-.2014E-01	-.1459E-01	-.1015E-01	-.7443E-02
STD	.3391	.2324	.1893	.1597	.1416	.1156	.9758E-01	.7874E-01
STD MEAN	.4796E-02	.4648E-02	.4646E-02	.4517E-02	.4479E-02	.4500E-02	.4364E-02	.4334E-02
SKEWNESS	-0.346	-0.044	-0.034	0.045	0.059	0.052	-0.011	-0.156
KURTOSIS	2.547	-0.074	-0.056	0.015	-0.014	-0.155	0.448	0.144

MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:	-.337710E-03	-1.02174	-.542526	-12.4196
VARIANCE OF REGRESSION - COEFFICIENTS:	.149923E-04	.141879E-01	7.25536	364.315
STD DEV OF REGRESSION - COEFFICIENTS:	.387199E-02	.119113	2.69358	19.0871
REGRESSION ON VARIANCE - COEFFICIENTS:	.713406	3.47347	-11.6751	15.9883

VERTICAL SCALE: YMIN = -0.9890
YMAX = 0.7503

ESTIMATOR: ESTIMATES OF THE Z-TRANSFORM OF THE SERIAL CORRELATION COEFFICIENT FOR A NORMAL(0,1) SAMPLE

Figure 4a. Estimates of the Z-Transform of the Serial Correlation Coefficient for a Normal (0,1) Sample



SUBSAMPLE SIZE	10	20	30	40	50	75	100	150
MEAN	-.1103	-.4744E-01	-.2791E-01	-.1800E-01	-.1231E-01	-.6246E-02	-.2488E-02	.1572E-02
STD	.2832	.2043	.1713	.1515	.1361	.1114	.9663E-01	.8421E-01
STD MEAN	.4005E-02	.4085E-02	.4204E-02	.4286E-02	.4303E-02	.4335E-02	.4321E-02	.4635E-02
SKEWNESS	0.146	0.561	0.631	0.534	0.497	0.435	0.513	0.394
KURTOSIS	0.474	0.327	0.470	0.362	0.100	-0.221	0.255	0.323

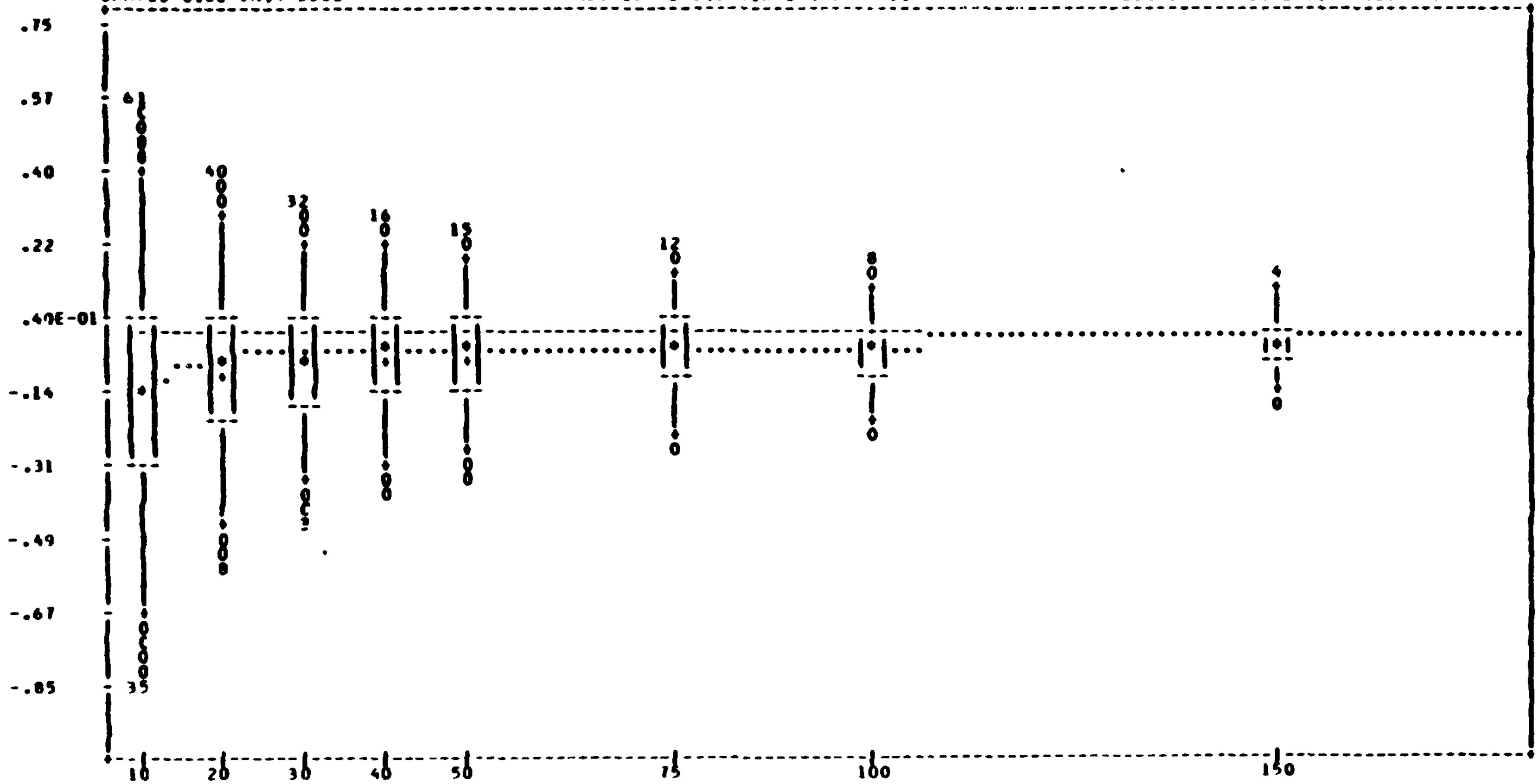
MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:	.770474E-02	-.955687	-3.68524	14.3363
VARIANCE OF REGRESSION - COEFFICIENTS:	.240493E-04	.138909E-01	10.6468	618.694
STD DEV OF REGRESSION - COEFFICIENTS:	.490401E-02	.117860	3.26203	24.8736
REGRESSION ON VARIANCE - COEFFICIENTS:	1.14227	-1.82462	1.51063	2.69722

VERTICAL SCALE: YMIN = -0.9890
YMAX = 0.7503

ESTIMATOR: ESTIMATES OF THE Z-TRANSFORM OF THE SERIAL CORRELATION COEFFICIENT FOR A CHI-SQUARE(1) SAMPLE

Figure 4b. Estimates of the Z-Transform of the Serial Correlation Coefficient for a Chi-Square (1) Sample

SAMPLE SIZE (N): 5000 NO. OF REPLICATIONS (M): 10 DEGREE OF REGRESSION (D): 3



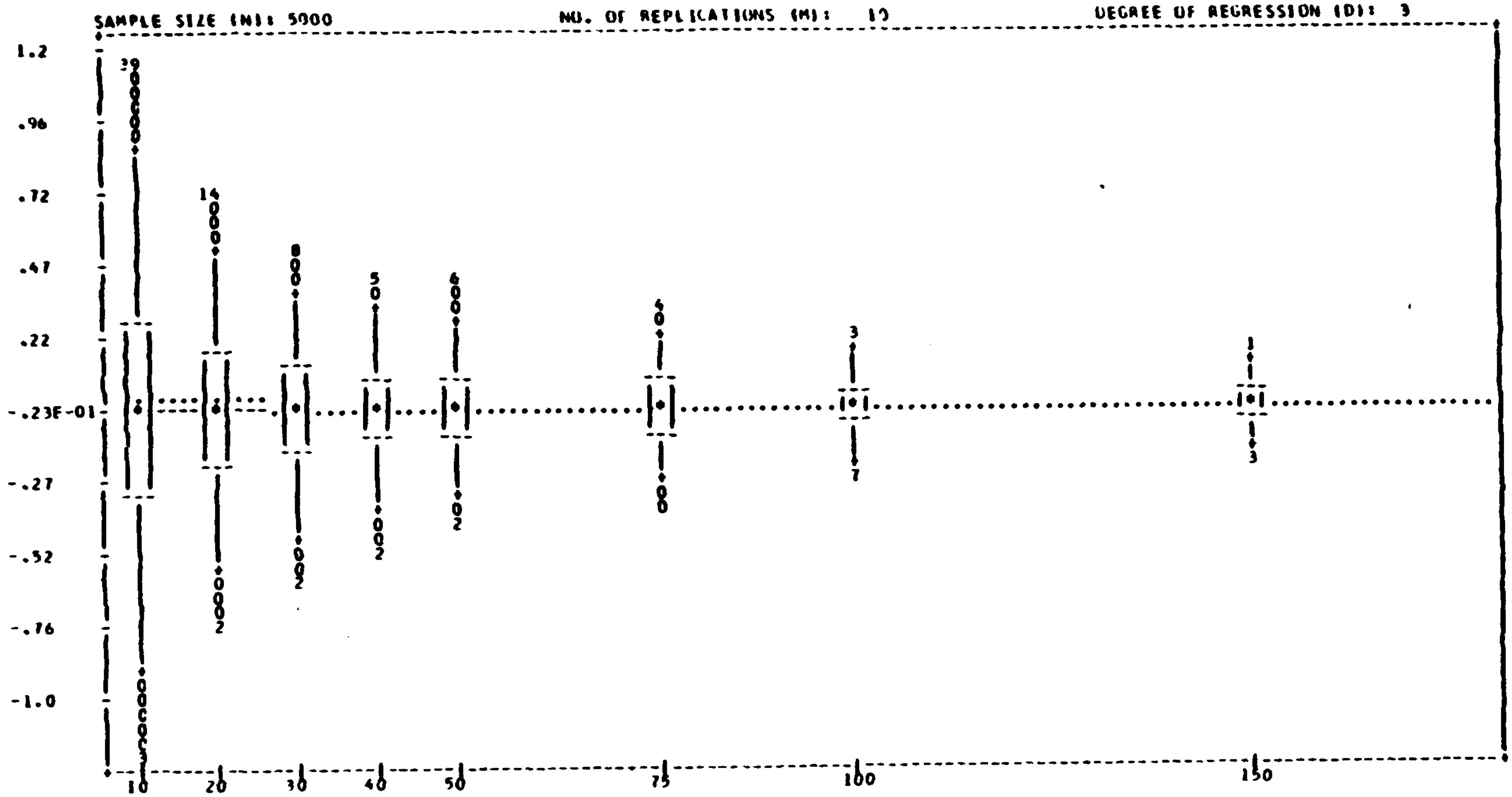
SUBSAMPLE SIZE	10	20	30	40	50	75	100	150
MEAN	-.1172	-.5411E-01	-.3724E-01	-.2857E-01	-.2316E-01	-.1695E-01	-.1268E-01	-.1046E-01
STD	.2849	.1970	.1619	.1362	.1216	.1004	.8603E-01	.6723E-01
STD MEAN	.4029E-02	.3941E-02	.3973E-02	.3852E-02	.3846E-02	.3909E-02	.3847E-02	.3701E-02
SKEWNESS	0.369	0.474	0.556	0.594	0.632	0.600	0.616	0.685
KURTOSIS	1.527	0.656	0.611	0.819	1.032	0.631	0.835	0.995

MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:	-.312200E-02	-1.03891	1.64581	-26.7689
VARIANCE OF REGRESSION - COEFFICIENTS:	.589428E-05	.425600E-01	22.7239	967.803
STD DEV OF REGRESSION - COEFFICIENTS:	.242782E-02	.206315	4.76696	31.1095
REGRESSION ON VARIANCE - COEFFICIENTS:	.545919	2.71325	-11.2125	16.7234

VERTICAL SCALE: YMIN = -0.9890
YMAX = 0.7503

ESTIMATOR: ESTIMATES OF THE Z-TRANSFORM OF THE SERIAL CORRELATION COEFFICIENT FOR A LOGNORMAL(0,1) SAMPLE

Figure 4c. Estimates of the Z-Transform of the Serial Correlation Coefficient for a Lognormal (0,1) Sample



SUBSAMPLE SIZE	10	20	30	40	50	75	100	150
MEAN	.1471E-01	.4979E-C2	.3881E-04	-.9342E-03	.2980E-02	-.9550E-03	-.3521E-C3	-.3150E-C3
STD	.4475	.2757	.2137	.1733	.1543	.1257	.1028	.8101E-01
STD MEAN	.6328E-02	.5513E-02	.5245E-02	.4901E-02	.4879E-02	.4892E-02	.4554E-C2	.4459E-02
SKEWNESS	0.319	0.190	0.125	0.130	0.152	0.134	-0.017	-0.126
KURTOSIS	0.094	0.008	-0.125	-0.031	0.059	-0.106	C.34C	0.045

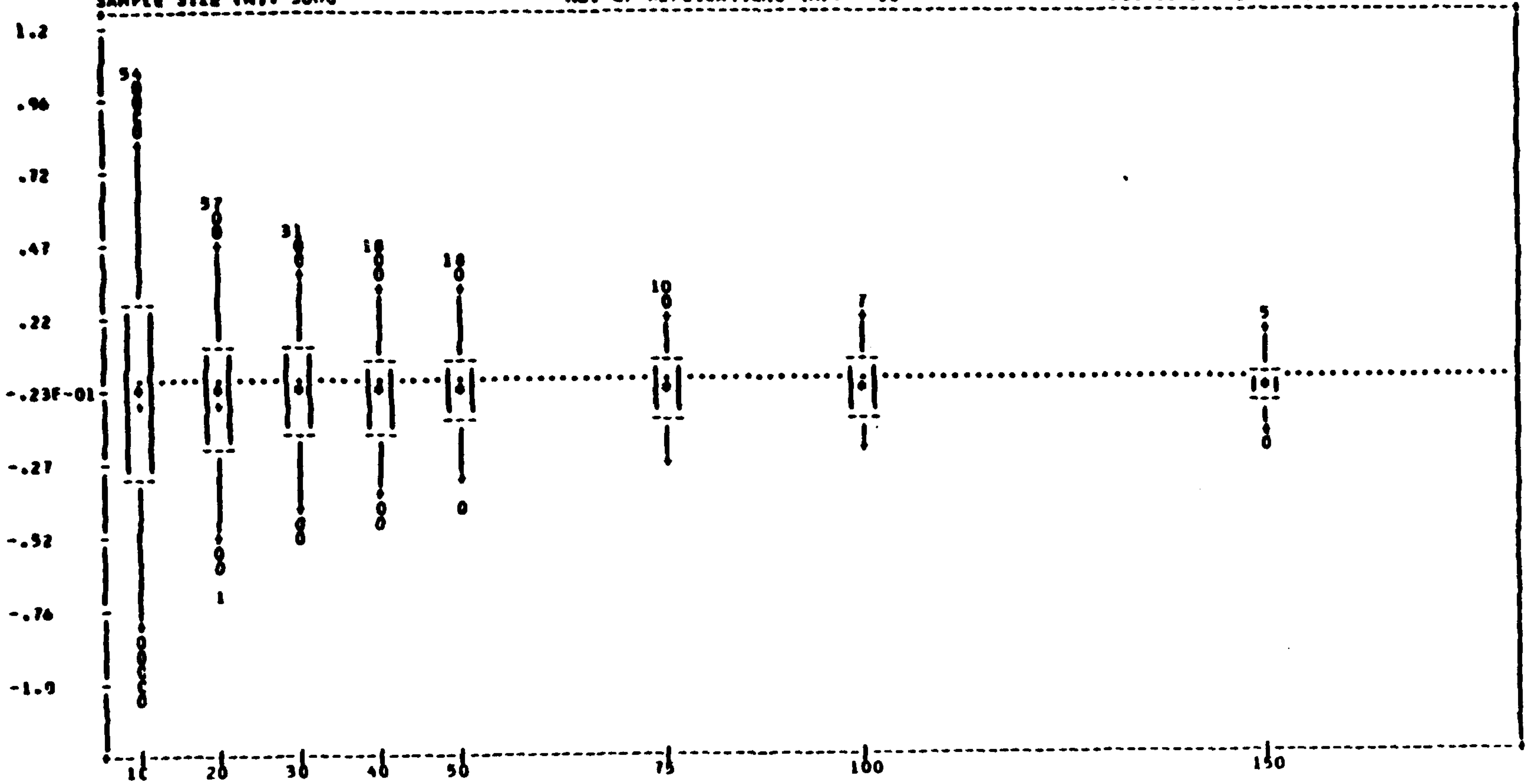
MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:	.546557E-02	-.131495	5.60890	-28.8484
VARIANCE OF REGRESSION - COEFFICIENTS:	.153676E-04	.906536E-01	89.6084	5054.53
STD DEV OF REGRESSION - COEFFICIENTS:	.392016E-02	.301087	9.46611	71.0952
REGRESSION ON VARIANCE - COEFFICIENTS:	.864258	1.28888	7.48415	-.560686

VERTICAL SCALE: YMIN = -1.2072
YMAX = 1.2114

ESTIMATOR: ESTIMATES OF THE 2-FOLD JACKNIFED SERIAL CORRELATION COEFFICIENT FOR A NORMAL(0,1) SAMPLE

Figure 5a. Estimates of the 2-Fold Jackknifed Serial Correlation Coefficient for a Normal (0,1) Sample

SAMPLE SIZE (N): 5000 NO. OF REPLICATIONS (M): 10 DEGREE OF REGRESSION (D): 3



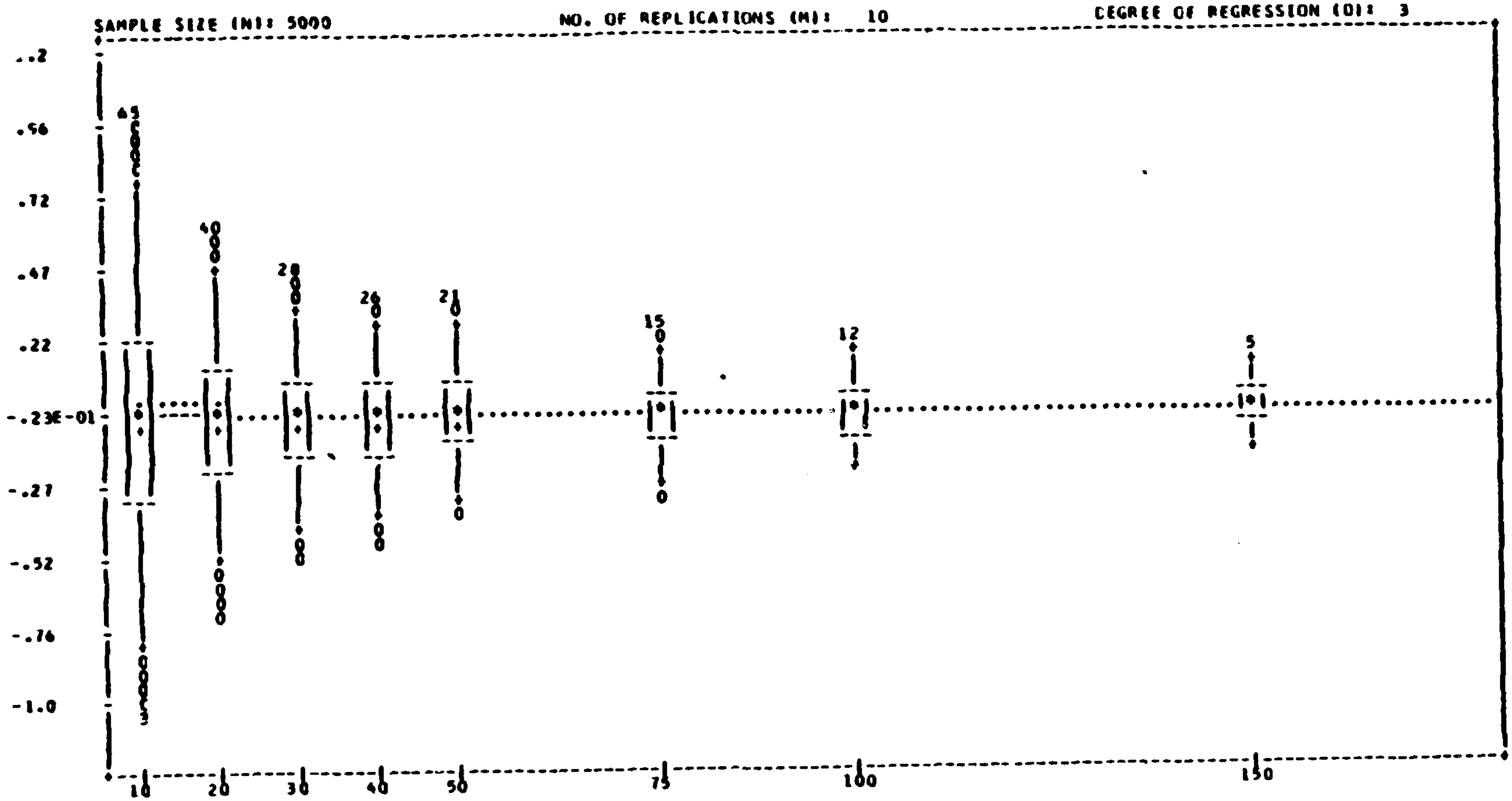
SUBSAMPLE SIZE	10	20	30	40	50	75	10C	150
MEAN	.2278E-01	.9513E-02	.1008E-01	.1067E-01	.1150E-01	.8868E-02	.7255E-C2	.9341E-02
STD	.4095	.2680	.2152	.1824	.1628	.1324	.1115	.9194E-01
STD MEAN	.5192E-02	.5361E-02	.5282E-02	.5159E-02	.5149E-02	.5154E-02	.5CC2E-C2	.5061E-02
SKEWNESS	0.544	0.804	0.813	0.683	0.694	0.724	C.5E5	0.581
KURTOSIS	0.138	0.701	0.888	0.424	0.444	0.44E	C.1E7	0.665

MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:	.578116E-02	.412461	-11.0824	86.4791
VARIANCE OF REGRESSION - COEFFICIENTS:	.270182E-04	.844226E-01	98.9C34	5462.96
STD DEV OF REGRESSION - COEFFICIENTS:	.520367E-02	.290556	9.94501	73.9118
REGRESSION ON VARIANCE - COEFFICIENTS:	1.11376	2.29230	-8.8444E	22.8625

VERTICAL SCALE: VMIN = -1.2072
VMAX = 1.2114

ESTIMATOR: ESTIMATES OF THE 2-FCLC JACKNIFED SERIAL CORRELATION COEFFICIENT FOR A CHI-SQUARE(1) SAMPLE

Figure 5b. Estimates of the 2-Fold Jackknifed Serial Correlation Coefficient for a Chi-Square (1) Sample



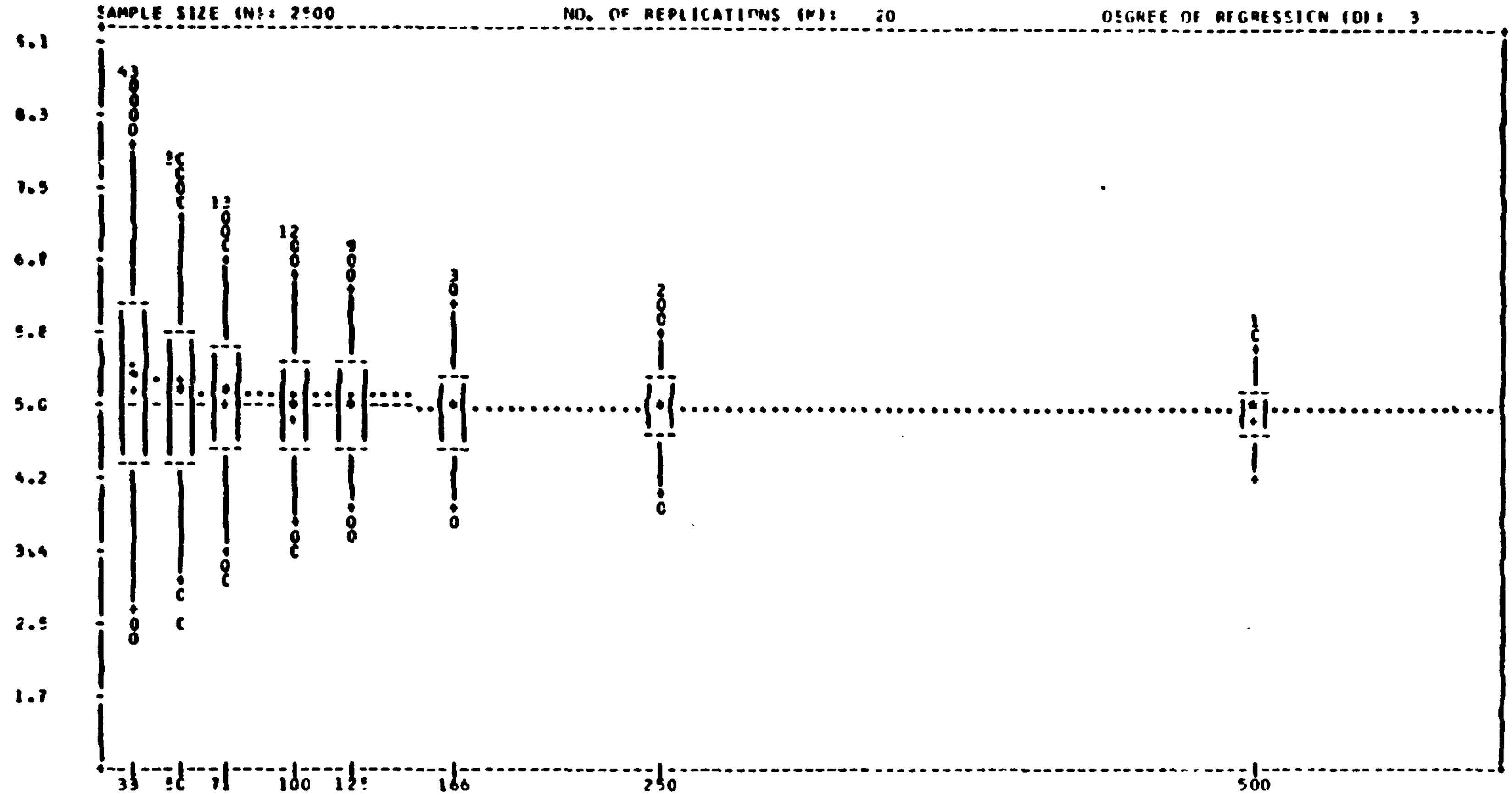
SUBSAMPLE SIZE	10	20	30	40	50	75	10C	150
MEAN	.1258E-01	.3435E-02	-.3343E-02	-.3821E-02	-.1695E-02	-.5042E-02	-.2255E-C2	-.3986E-02
STD	.4028	.2600	.2011	.1724	.1533	.1222	.1011	.7999E-01
STD MEAN	.5696E-02	.5200E-02	.4936E-02	.4876E-02	.4849E-02	.4759E-02	.4519E-C2	.4403E-02
SKEWNESS	0.550	0.712	0.741	0.831	0.894	0.819	C.744	0.850
KURTOSIS	0.352	0.662	0.905	1.077	1.602	1.15C	C.625	1.451

MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:	-.134541E-C2	-.337504	12.0957	-73.3541
VARIANCE OF REGRESSION - COEFFICIENTS:	.167209E-04	.218834	154.825	7199.73
STD DEV OF REGRESSION - COEFFICIENTS:	.408912E-C2	.467797	12.4429	84.8512
REGRESSION ON VARIANCE - COEFFICIENTS:	.617429	5.83375	-19.4923	35.1018

VERTICAL SCALE: YMIN = -1.2072
YMAX = 1.2114

ESTIMATOR: ESTIMATES OF THE 2-FCLC JACKNIFED SERIAL CORRELATION COEFFICIENT FOR A LOGNORMAL(C,1) SAMPLE

Figure 5c. Estimates of the 2-Fold Jackknifed Serial Correlation Coefficient for a Lognormal (0,1) Sample



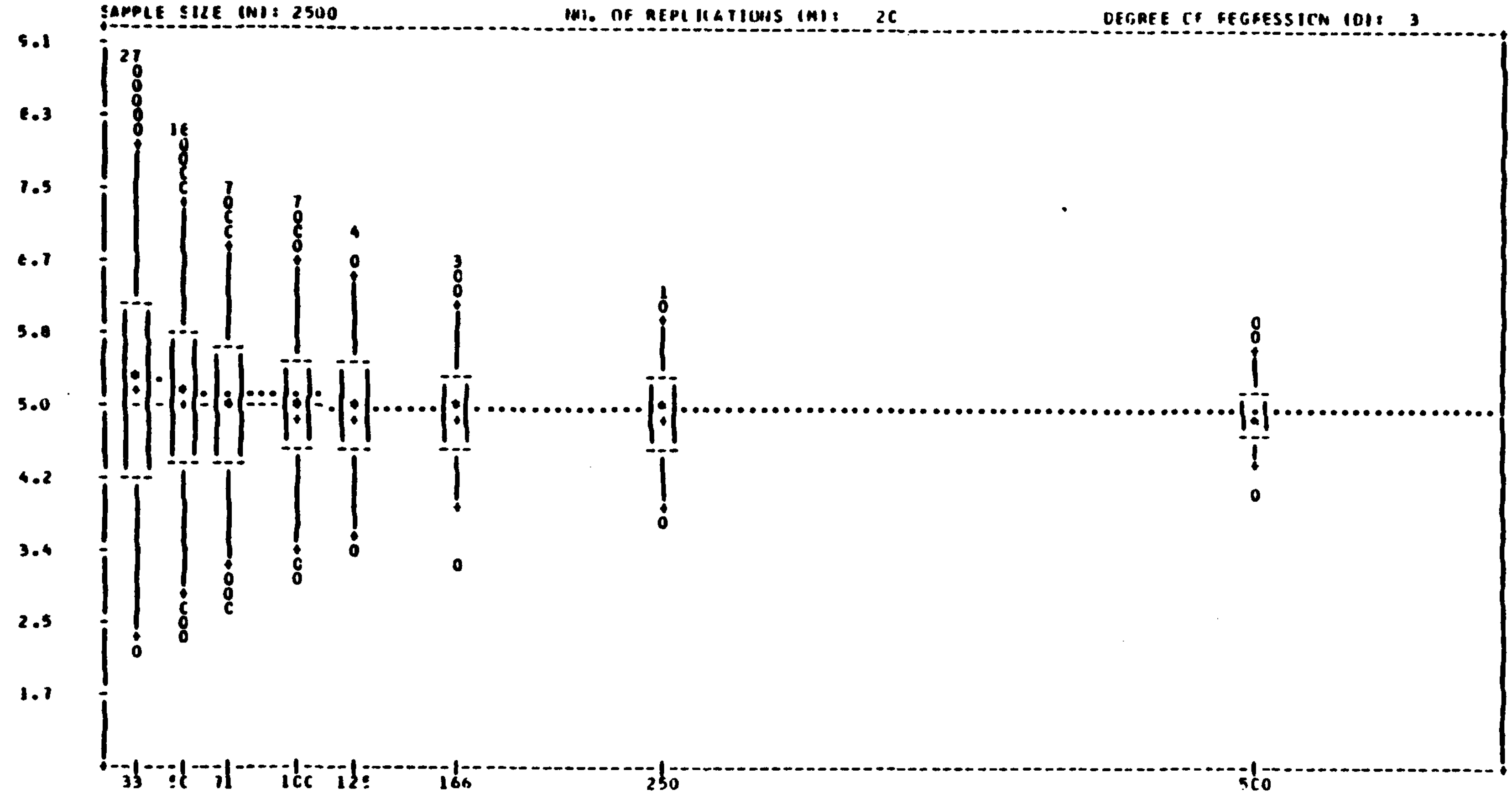
SUBSAMPLE SIZE	33	50	71	100	125	166	250	500
MEAN	5.477	5.309	5.202	5.145	5.118	5.085	5.045	5.026
STD	1.440	1.137	.8725	.7671	.6650	.5622	.4807	.3654
STD MEAN	.3738E-01	.3595E-01	.3298E-01	.3431E-01	.3325E-01	.3246E-01	.3399E-01	.3654E-01
SKEWNESS	1.217	1.074	0.732	0.812	0.680	0.525	0.354	0.679
KURTOSIS	2.872	2.156	0.904	0.762	0.653	0.157	0.246	0.119

MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:	4.99635	13.5123	140.737	-2063.17
VARIANCE OF REGRESSION - COEFFICIENTS:	.110506E-02	26.9280	158094.	.579950E+08
STD DEV OF REGRESSION - COEFFICIENTS:	.332424E-01	6.07684	397.610	7615.39
REGRESSION ON VARIANCE - COEFFICIENTS:	116.458	-1681.96	14121.1	-34565.3

VERTICAL SCALE: YMIN = 1.0544
YMAX = 9.1332

ESTIMATOR: MAXIMUM LIKELIHOOD ESTIMATE OF THE SHAPE PARAMETER OF THE GAMMA DISTRIBUTION K=5.

Figure 6a. Maximum Likelihood Estimate of the Shape Parameter of the Gamma Distribution (k = 5.0)



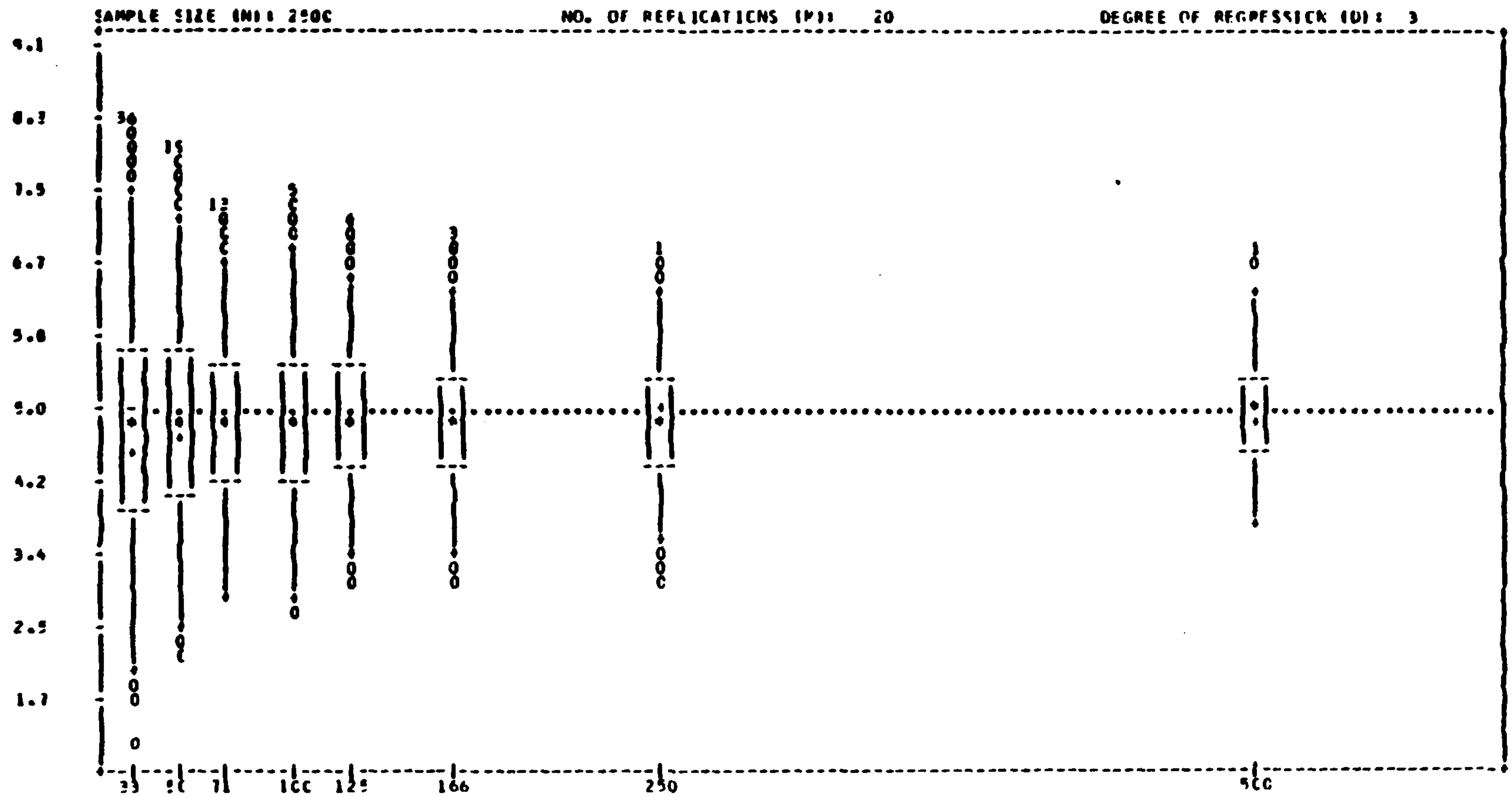
SUBSAMPLE SIZE	33	50	71	100	125	166	250	500
MEAN	5.371	5.234	5.151	5.113	5.080	5.054	5.028	4.997
STD	1.482	1.177	.9606	.8281	.7313	.6277	.5268	.3750
STD MEAN	.3827E-01	.3723E-01	.2631E-01	.3703E-01	.3657E-01	.3624E-01	.3725E-01	.3750E-01
SKEWNESS	0.858	0.702	0.591	0.572	0.525	0.429	0.354	0.482
KURTOSIS	1.512	1.005	0.670	0.375	0.373	-0.184	-0.120	-0.135

MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:	4.96799	15.4346	-196.572	4202.59
VARIANCE OF REGRESSION - COEFFICIENTS:	.966283E-03	4.14804	23214.0	.989117E+07
STD DEV OF REGRESSION - COEFFICIENTS:	.310851E-01	2.03667	152.361	3145.01
REGRESSION ON VARIANCE - COEFFICIENTS:	81.7175	-336.834	2162.00	-3040.97

VERTICAL SCALE: YMIN = 1.0544
YMAX = 9.1332

ESTIMATOR: MOMENT ESTIMATOR (RECIPROCAL OF SQUARED COEFFICIENT OF VARIATION) OF THE SHAPE PARAMETER OF GAMMA DISTRIBUTION K=5.

Figure 6b. Moment Estimator (Reciprocal of Squared Coefficient of Variation) of the Shape Parameter of the Gamma Distribution (k = 5.0)



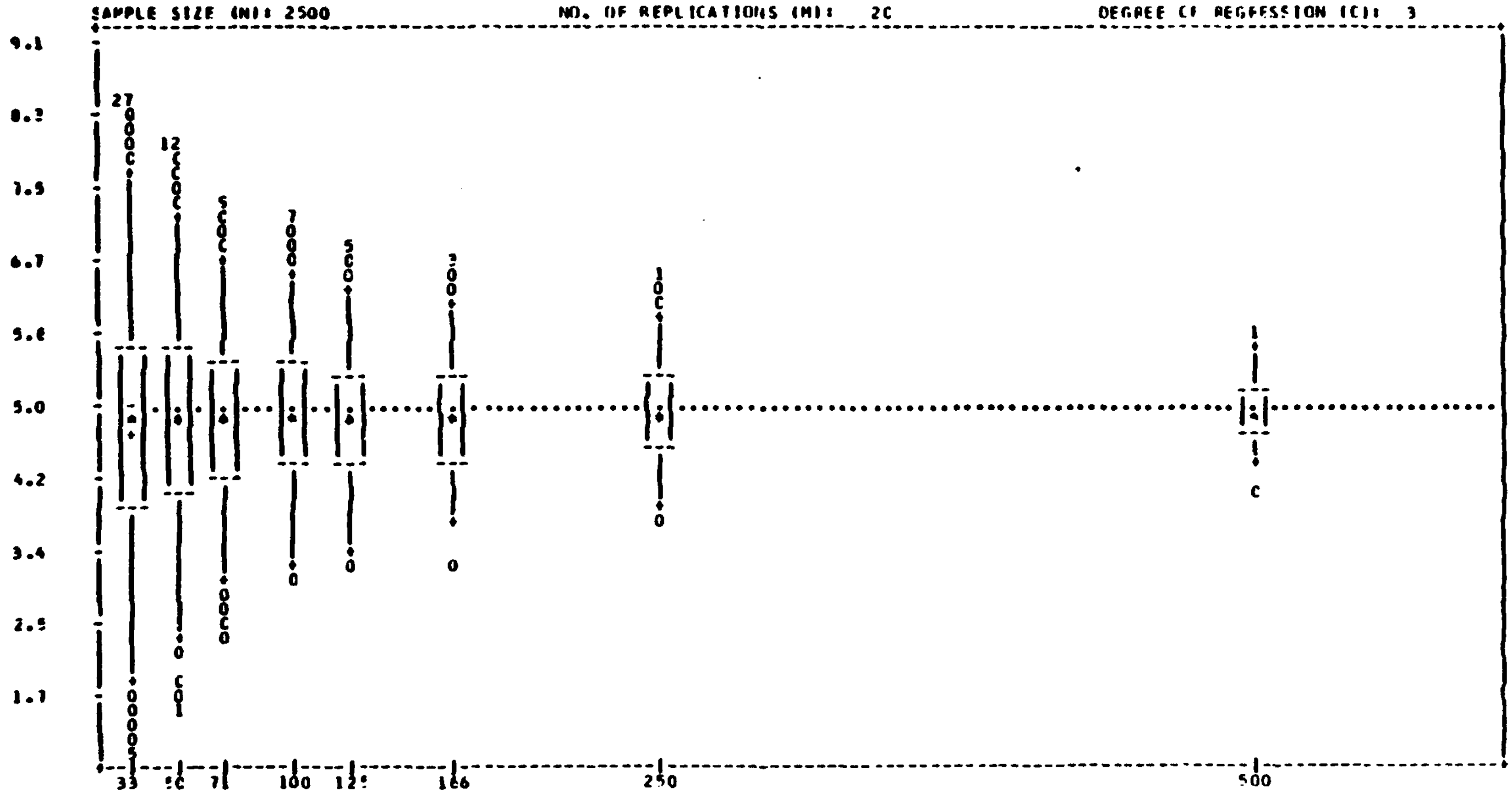
SUBSAMPLE SIZE	33	50	71	100	125	166	250	500
MEAN	4.517	4.964	4.570	4.577	5.008	5.010	4.968	5.031
STD	1.520	1.241	1.012	.9542	.8517	.7624	.7210	.6994
STD MEAN	.3525E-01	.3921E-01	.3826E-01	.4267E-01	.4259E-01	.4402E-01	.5098E-01	.6854E-01
SKEWNESS	1.177	0.960	0.912	0.616	0.524	0.364	-0.045	0.465
KURTOSIS	2.840	1.823	1.785	0.230	0.361	-0.155	0.018	-0.244

MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:	5.02893	-7.10724	349.841	-7711.72
VARIANCE OF REGRESSION - COEFFICIENTS:	.103702E-01	550.288	.257296E+07	.937448E+09
STD DEV OF REGRESSION - COEFFICIENTS:	.101834	24.2958	1604.05	30617.6
REGRESSION ON VARIANCE - COEFFICIENTS:	587.234	-11703.4	87638.4	-214209.

VERTICAL SCALE: YMIN = 1.0544
YMAX = 9.1332

ESTIMATOR: 4-FOLD JACKKNIFED MAXIMUM LIKELIHOOD ESTIMATE OF THE SHAPE PARAMETER OF THE GAMMA DISTRIBUTION K=5.

Figure 6c. 4-Fold Jackknifed Maximum Likelihood Estimate of the Shape Parameter of the Gamma Distribution (k = 5.0)



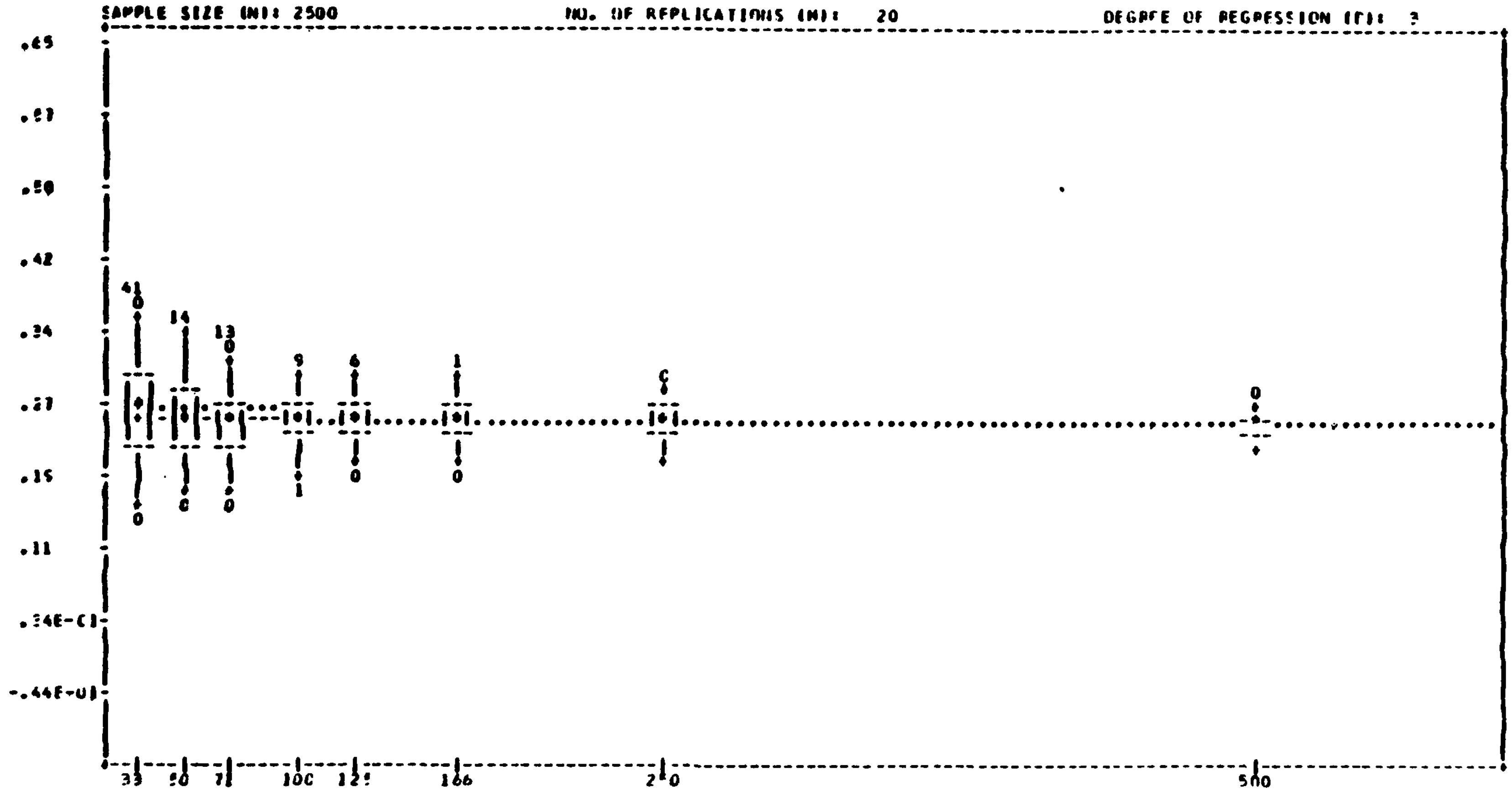
SUBSAMPLE SIZE	33	50	71	100	125	166	250	500
MEAN	4.919	4.951	4.958	4.984	4.977	4.980	4.975	4.965
STD	1.530	1.211	.9855	.8476	.7416	.6342	.5298	.3734
STD MEAN	.3549E-01	.3853E-01	.3725E-01	.3751E-01	.3708E-01	.3662E-01	.3746E-01	.3734E-01
SKEWNESS	0.651	0.464	0.524	0.517	0.515	0.437	0.323	0.441
KURTOSIS	1.969	0.510	0.738	0.376	0.376	-0.162	-0.132	-0.166

MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:	4.96213	5.41940	-472.241	8152.19
VARIANCE OF REGRESSION - COEFFICIENTS:	.905102E-03	13.9244	91080.1	.426571E+08
STD DEV OF REGRESSION - COEFFICIENTS:	.301513E-01	3.73152	303.117	6531.21
REGRESSION ON VARIANCE - COEFFICIENTS:	81.2402	-382.332	3397.67	-7671.09

VERTICAL SCALE: YMIN = 1.0546
YMAX = 9.1312

ESTIMATOR: 4-FOLD JACKKNIFED MOMENT ESTIMATOR OF THE SHAPE PARAMETER OF THE GAMMA DISTRIBUTION K=5.

Figure 6d. 4-Fold Jackknifed Moment Estimator of the Shape Parameter of the Gamma Distribution (k = 5.0)



SUBSAMPLE SIZE	33	50	71	100	125	166	250	500
MEAN	.2683	.2625	.2596	.2570	.2561	.2554	.2542	.2533
STD	.5589E-01	.4315E-01	.3680E-01	.2865E-01	.2595E-01	.2304E-01	.1821E-01	.1255E-01
STD MEAN	.1443E-02	.1366E-02	.1391E-02	.1281E-02	.1298E-02	.1330E-02	.1288E-02	.1255E-02
SKEWNESS	1.015	1.750	0.741	0.408	0.448	0.326	0.171	0.017
KURTOSIS	1.771	4.950	0.709	-0.035	0.235	0.170	-0.301	-0.120

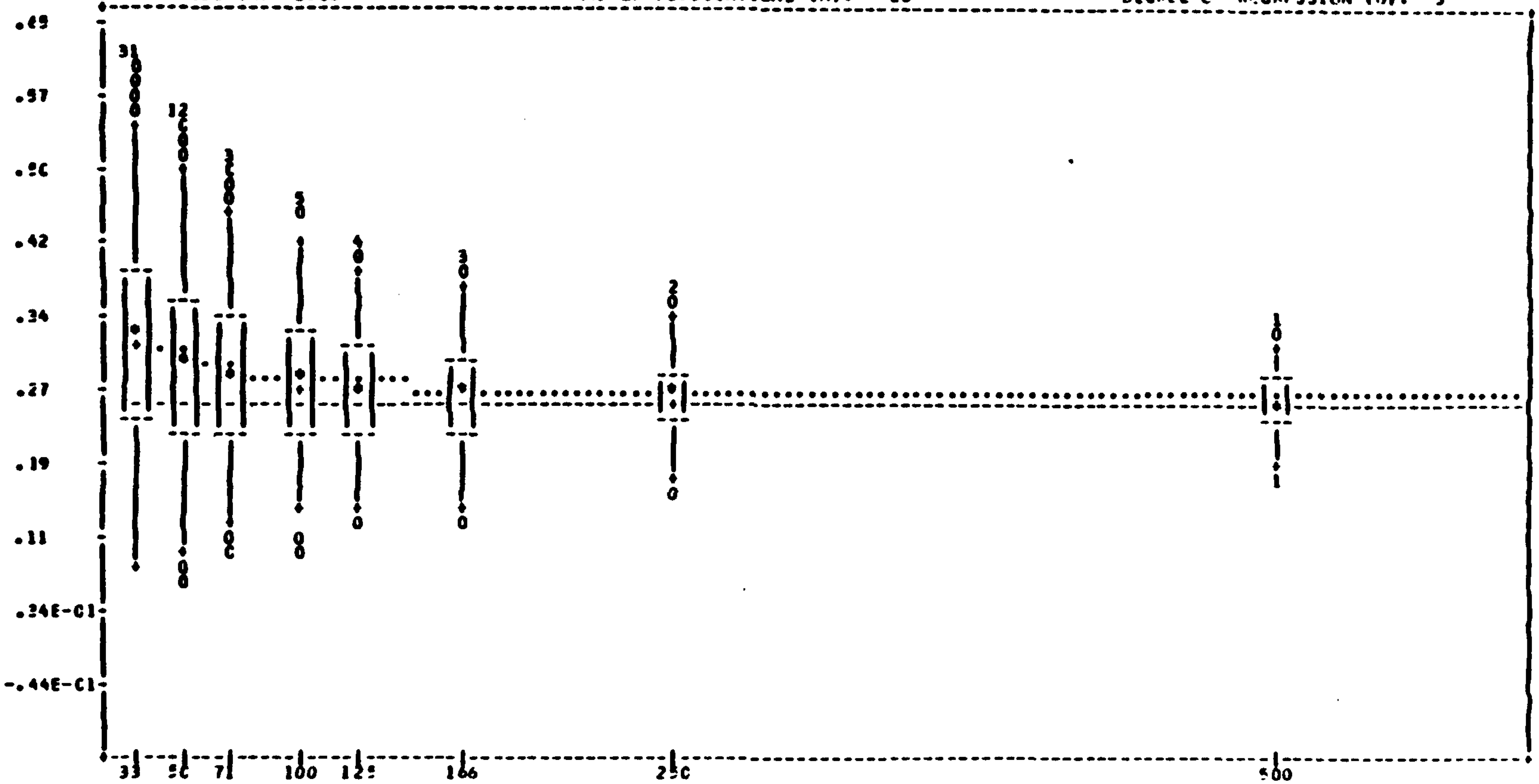
MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:	.252401	.445971	4.43231	-60.8146
VARIANCE OF REGRESSION - COEFFICIENTS:	.158013E-05	.662472E-02	40.5503	18732.8
STD DEV OF REGRESSION - COEFFICIENTS:	.140717E-02	.813524E-01	6.36791	136.868
REGRESSION ON VARIANCE - COEFFICIENTS:	.733840E-01	.161415	-.519010	3.25439

VERTICAL SCALE: YMIN = -0.1055
YMAX = 0.6520

ESTIMATOR: MAXIMUM LIKELIHOOD ESTIMATE OF THE SHAPE PARAMETER OF THE GAMMA DISTRIBUTION K=.25

Figure 7a. Maximum Likelihood Estimate of the Shape Parameter of the Gamma Distribution (k = 0.25)

SAMPLE SIZE (N): 2500 NO. OF REPLICATIONS (M): 20 DEGREE OF REGRESSION (D): 3



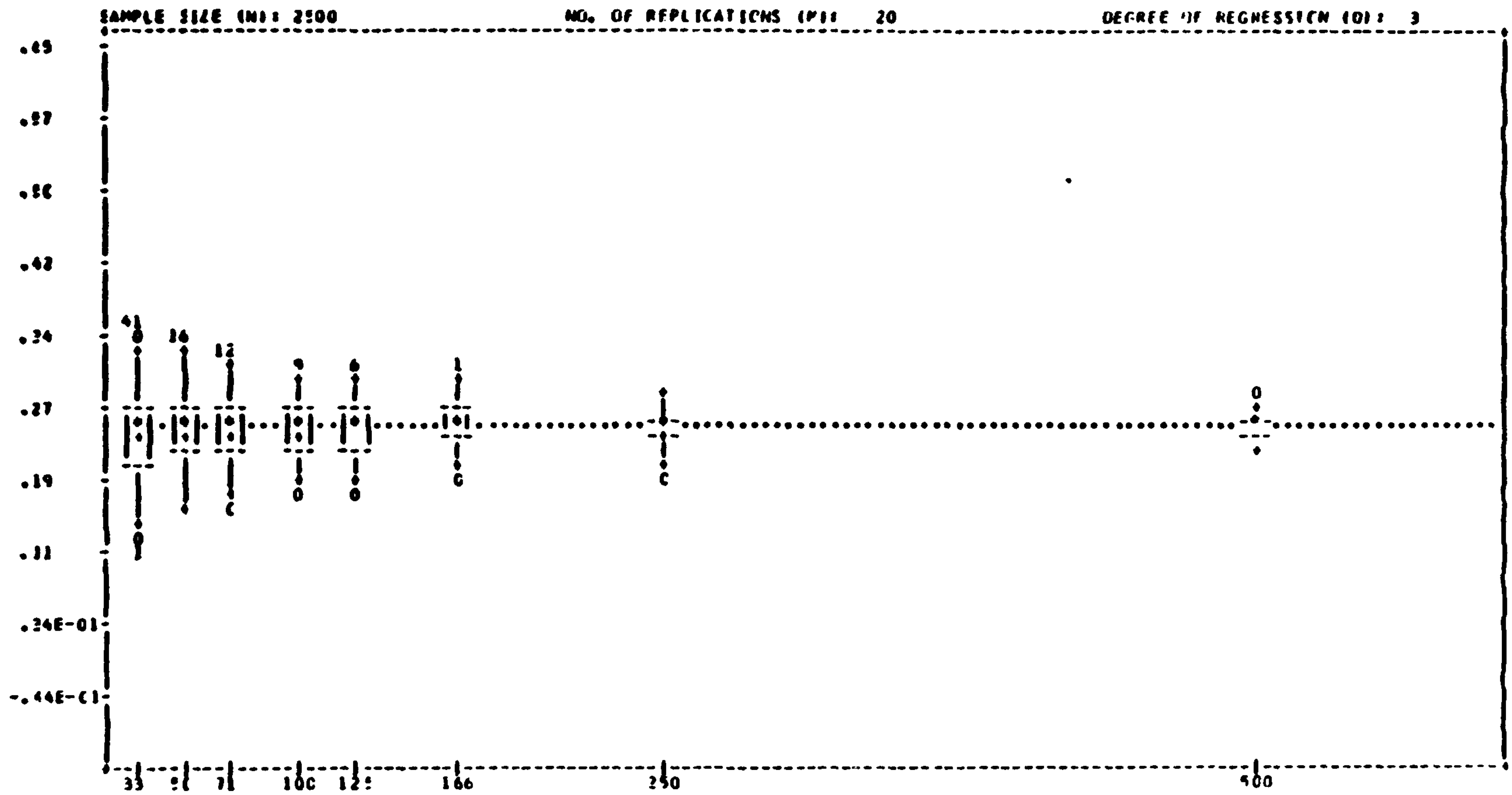
SUBSAMPLE SIZE	33	50	71	100	125	166	250	500
MEAN	.3288	.3064	.2512	.2814	.2759	.2712	.2658	.2600
STD	.1209	.9821E-01	.8061E-01	.7022E-01	.6153E-01	.5624E-01	.4595E-01	.3546E-01
STD MEAN	.3123E-02	.3101E-02	.3047E-02	.3140E-02	.3096E-02	.3247E-02	.3252E-02	.3546E-02
SKEWNESS	0.765	0.565	0.345	0.399	0.351	0.254	0.355	0.362
KURTOSIS	0.746	0.347	-0.117	0.467	0.200	-0.106	0.340	0.728

MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:	.254517	2.67586	-1.86707	-193.781
VARIANCE OF REGRESSION - COEFFICIENTS:	.970740E-05	.211753	874.454	364349.
STD DEV OF REGRESSION - COEFFICIENTS:	.311567E-02	.460167	29.5712	[illegible]
REGRESSION ON VARIANCE - COEFFICIENTS:	.974941	-11.0911	80.0564	-187.121

VERTICAL SCALE: YMIN = -0.1055
YMAX = 0.6520

ESTIMATOR: MOMENT ESTIMATOR (RECIPROCAL OF SQUARED COEFFICIENT OF VARIATION) OF THE SHAPE PARAMETER OF GAMMA DISTRIBUTION K=.25

Figure 7b. Moment Estimator (Reciprocal of Squared Coefficient of Variation) of the Shape Parameter of the Gamma Distribution (k = 0.25)



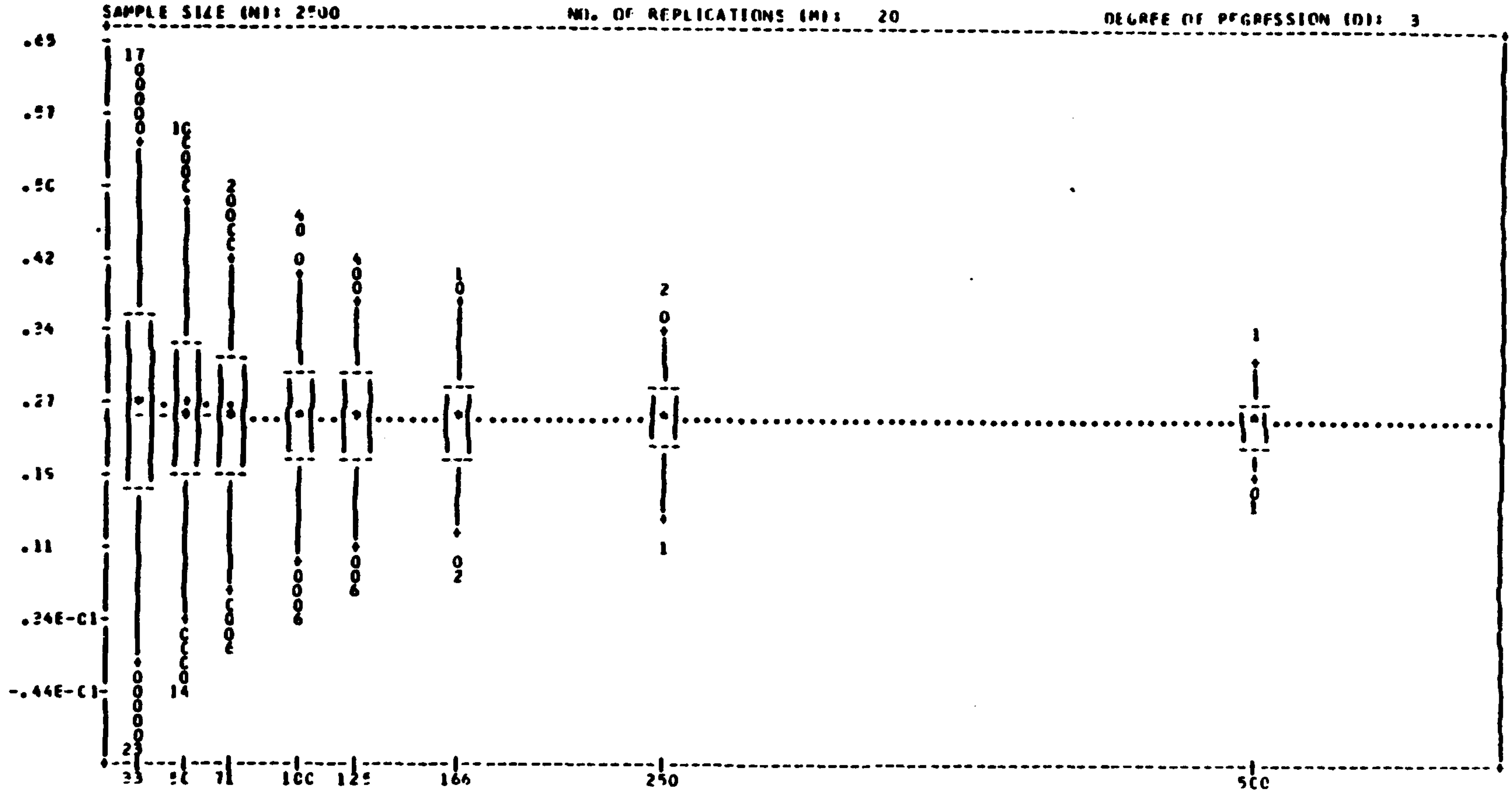
SUBSAMPLE SIZE	33	50	71	100	125	166	250	500
MEAN	.2506	.2511	.2521	.2518	.2520	.2523	.2523	.2524
STD	.6316E-01	.4151E-01	.3564E-01	.2809E-01	.2575E-01	.2250E-01	.1818E-01	.1264E-01
STD MEAN	.1373E-02	.1311E-02	.1347E-02	.1256E-02	.1287E-02	.1322E-02	.1298E-02	.1264E-02
SKEWNESS	0.996	0.736	0.736	0.413	0.435	0.327	0.170	0.049
KURTOSIS	1.602	0.982	0.712	0.069	0.270	0.163	-0.322	-0.045

MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:	.252554	-.524509E-01	4.61397	-122.667
VARIANCE OF REGRESSION - COEFFICIENTS:	.208554E-05	.181097E-01	116.478	54580.3
STD DEV OF REGRESSION - COEFFICIENTS:	.144414E-02	.134572	10.7925	233.624
REGRESSION ON VARIANCE - COEFFICIENTS:	.738654E-01	.237421	-2.13279	8.05335

VERTICAL SCALE: YMIN = -0.1055
YMAX = 0.6570

ESTIMATOR: 4-FOLD JACKKNIFED MAXIMUM LIKELIHOOD ESTIMATE OF THE SHAPE PARAMETER OF THE GAMMA DISTRIBUTION K=.25

Figure 7c. 4-Fold Jackknifed Maximum Likelihood Estimate of the Shape Parameter of the Gamma Distribution (k = 0.25)



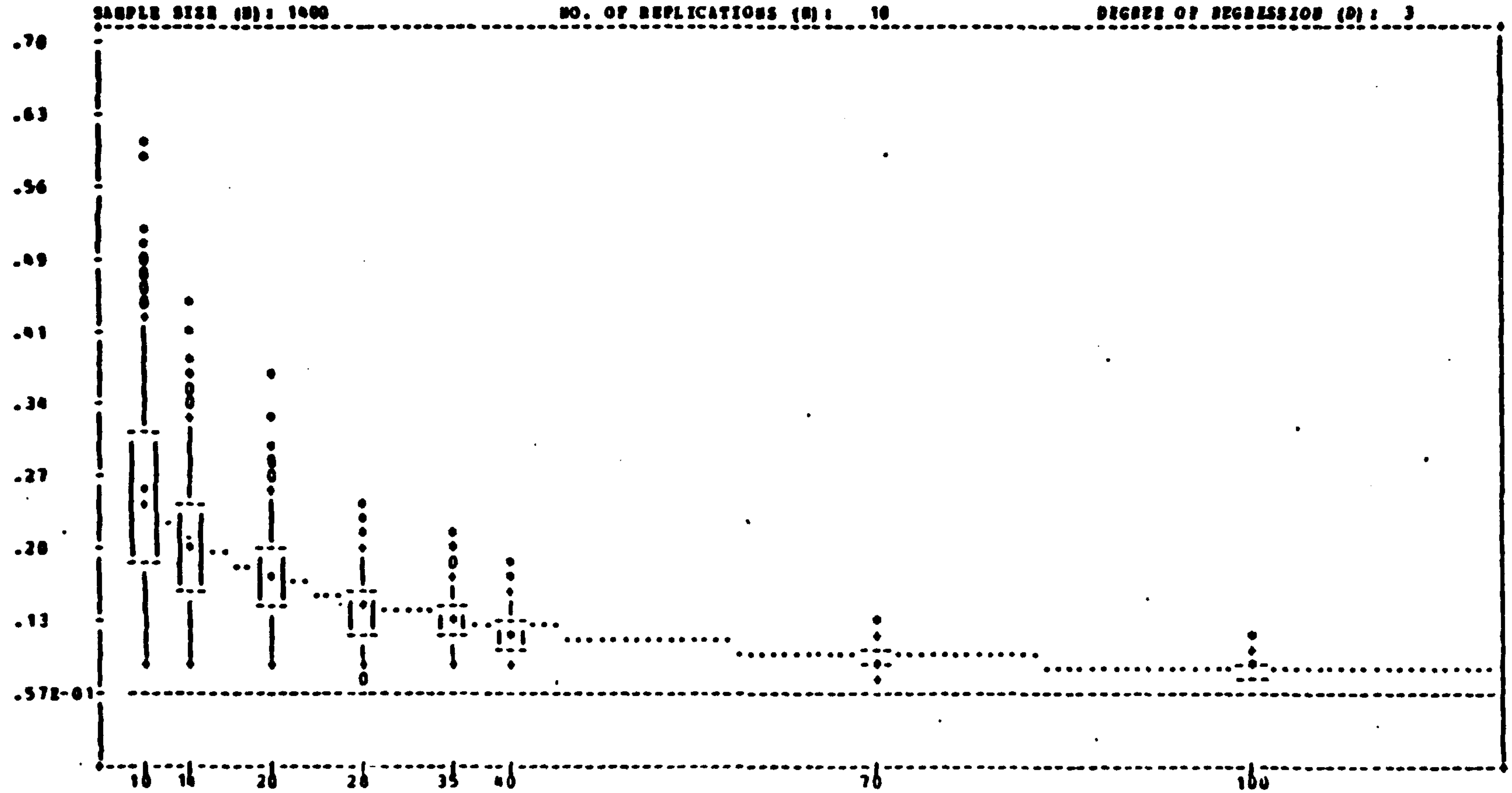
SUBSAMPLE SIZE	33	50	71	100	125	166	250	500
MEAN	.2685	.2623	.2579	.2561	.2546	.2549	.2555	.2541
STD	.1594	.1225	.9679E-01	.8482E-01	.7370E-01	.6461E-01	.5092E-01	.3773E-01
STD MEAN	.4115E-02	.3883E-02	.3650E-02	.3793E-02	.3685E-02	.3731E-02	.3601E-02	.3773E-02
SKEWNESS	-0.150	-0.154	-0.204	-0.136	-0.103	-0.153	0.171	-0.014
KURTOSIS	0.552	0.550	0.451	0.684	0.435	0.401	0.301	1.329

MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:	.255242	-.374536	53.4121	-877.151
VARIANCE OF REGRESSION - COEFFICIENTS:	.146565E-04	.483583	3064.30	.152388E+07
STD DEV OF REGRESSION - COEFFICIENTS:	.382838E-02	.695401	55.3561	1234.46
REGRESSION ON VARIANCE - COEFFICIENTS:	.750446	-1.48800	4.12576	42.7654

VERTICAL SCALE: YMIN = -0.1055
YMAX = 0.6530

ESTIMATOR: 4-FOLD JACKKNIFED MOMENT ESTIMATOR OF THE SHAPE PARAMETER OF THE GAMMA DISTRIBUTION K=.25

Figure 7d. 4-Fold Jackknifed Moment Estimator of the Shape Parameter of the Gamma Distribution (k = 0.25)



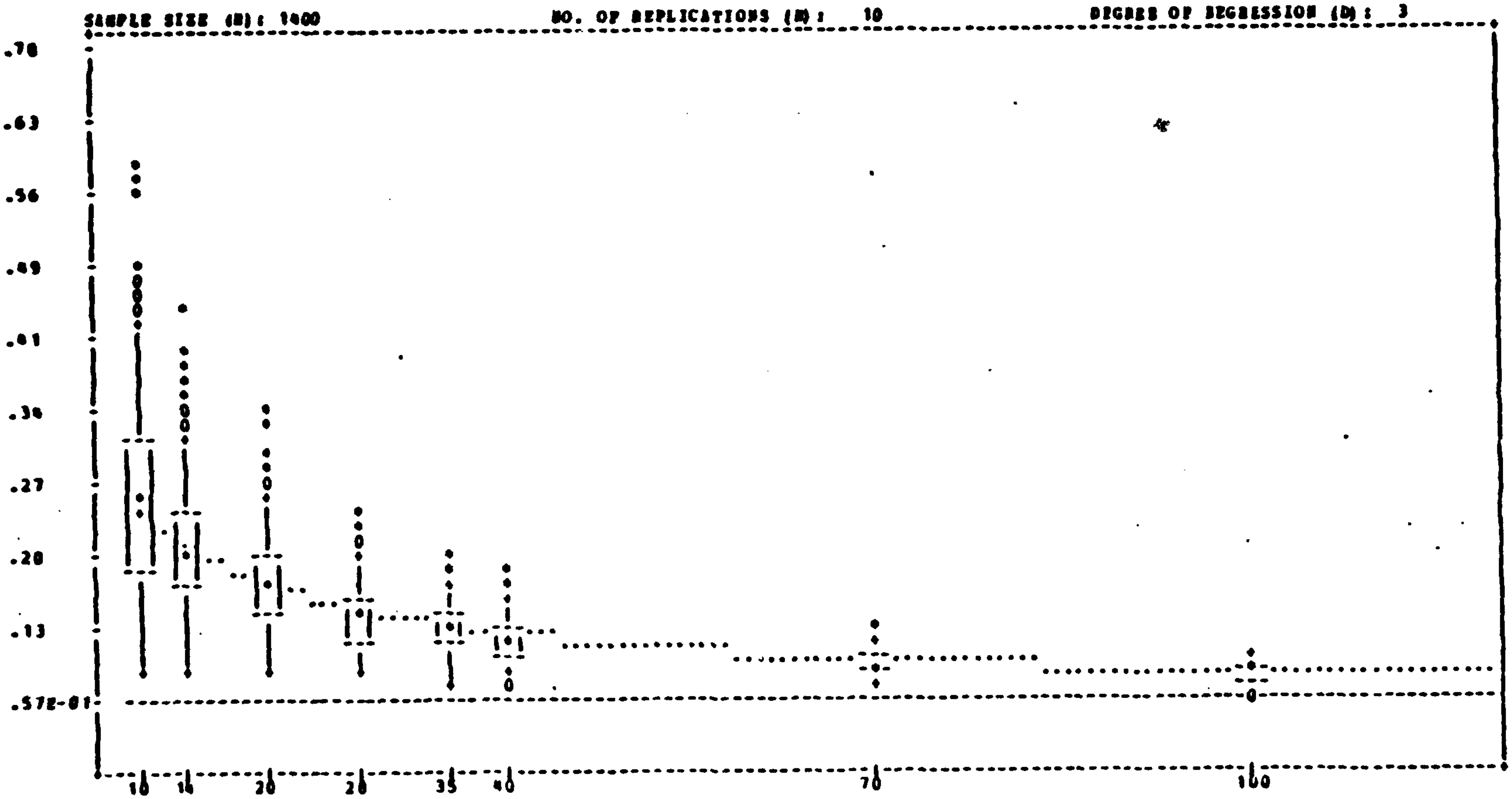
SUBSAMPLE SIZE	10	14	20	28	35	40	70	100
MEAN	.2588	.2119	.1758	.1500	.1338	.1267	.9912E-01	.8698E-01
S.D	.8279E-01	.5852E-01	.4197E-01	.3000E-01	.2165E-01	.2014E-01	.1144E-01	.7635E-02
STD MEAN	.2213E-02	.1850E-02	.1586E-02	.1342E-02	.1183E-02	.1076E-02	.8087E-03	.6452E-03
SKEWNESS	0.322	0.406	0.580	0.504	0.431	0.479	0.838	0.530
KURTOSIS	-0.679	0.066	0.711	0.109	-0.022	-0.165	0.829	0.307

MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:	.559309E-01	3.34941	-24.3771	111.744
VARIANCE OF REGRESSION - COEFFICIENTS:	.703211E-06	.288120E-02	2.53589	152.439
STD DEV OF REGRESSION - COEFFICIENTS:	.838576E-03	.536768E-01	1.59245	12.3466
REGRESSION ON VARIANCE - COEFFICIENTS:	-.146940E-01	.213476	-.292067	1.41739

VERTICAL SCALE: YMIN = 0.0
YMAX = 0.7000

ESTIMATOR: ESTIMATES OF THE STANDARD DEVIATION OF THE CORRELATION COEFFICIENT FOR A BIVARIATE NORMAL (0,1,0.5) BOOTSTRAP B=128

Figure 8. Estimates of the Standard Deviation of the Correlation Coefficient for a Bivariate Normal (0,1,0.5) Bootstrap B = 128



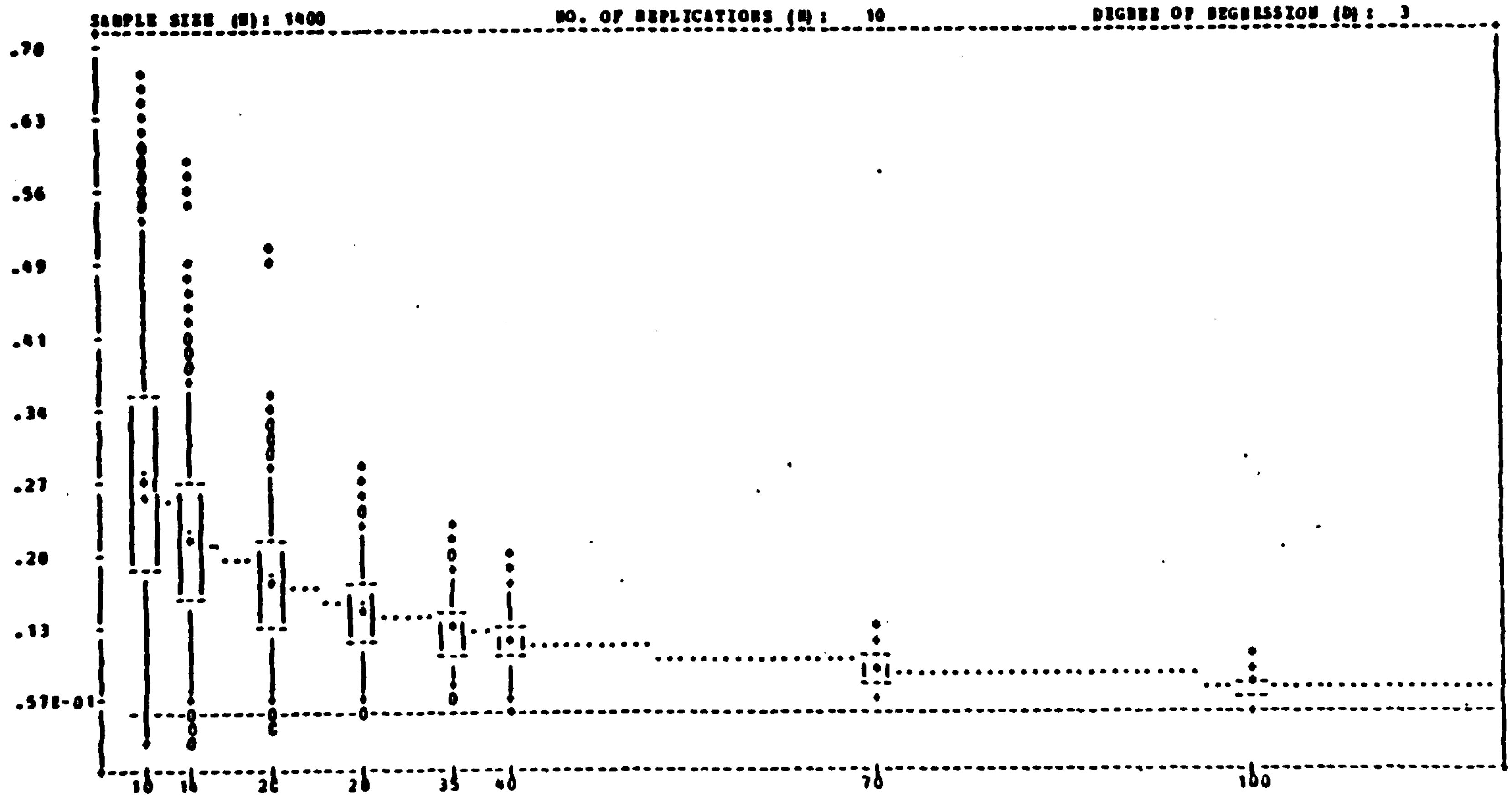
SUBSAMPLE SIZE	10	14	20	28	35	40	70	100
MEAN	.2589	.2122	.1764	.1509	.1341	.1271	.9920E-01	.8710E-01
STD	.6244E-01	.5778E-01	.4157E-01	.3095E-01	.2381E-01	.2069E-01	.1025E-01	.7765E-02
STD MEAN	.2403E-02	.1827E-02	.1571E-02	.1384E-02	.1191E-02	.1106E-02	.7247E-03	.6512E-03
SKEWNESS	0.280	0.420	0.567	0.544	0.532	0.461	0.284	0.219
KURTOSIS	-0.145	0.243	0.510	0.120	0.230	0.115	0.082	-0.657

MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:	.554341E-01	3.41007	-25.3869	116.362
VARIANCE OF REGRESSION - COEFFICIENTS:	.368017E-06	.258670E-02	2.87652	150.951
STD DEV OF REGRESSION - COEFFICIENTS:	.606644E-03	.508597E-01	1.69603	12.2863
REGRESSION ON VARIANCE - COEFFICIENTS:	-.353872E-01	.543053	-1.90031	3.84290

VERTICAL SCALE: THIN = 0.0
THAX = 0.7000

ESTIMATOR: ESTIMATES OF THE STANDARD DEVIATION OF THE CORRELATION COEFFICIENT FOR A BIVARIATE NORMAL (0,1,0.5) BOOTSTRAP B=512

Figure 9. Estimates of the Standard Deviation of the Correlation Coefficient for a Bivariate Normal (0,1,0.5) Bootstrap B = 512



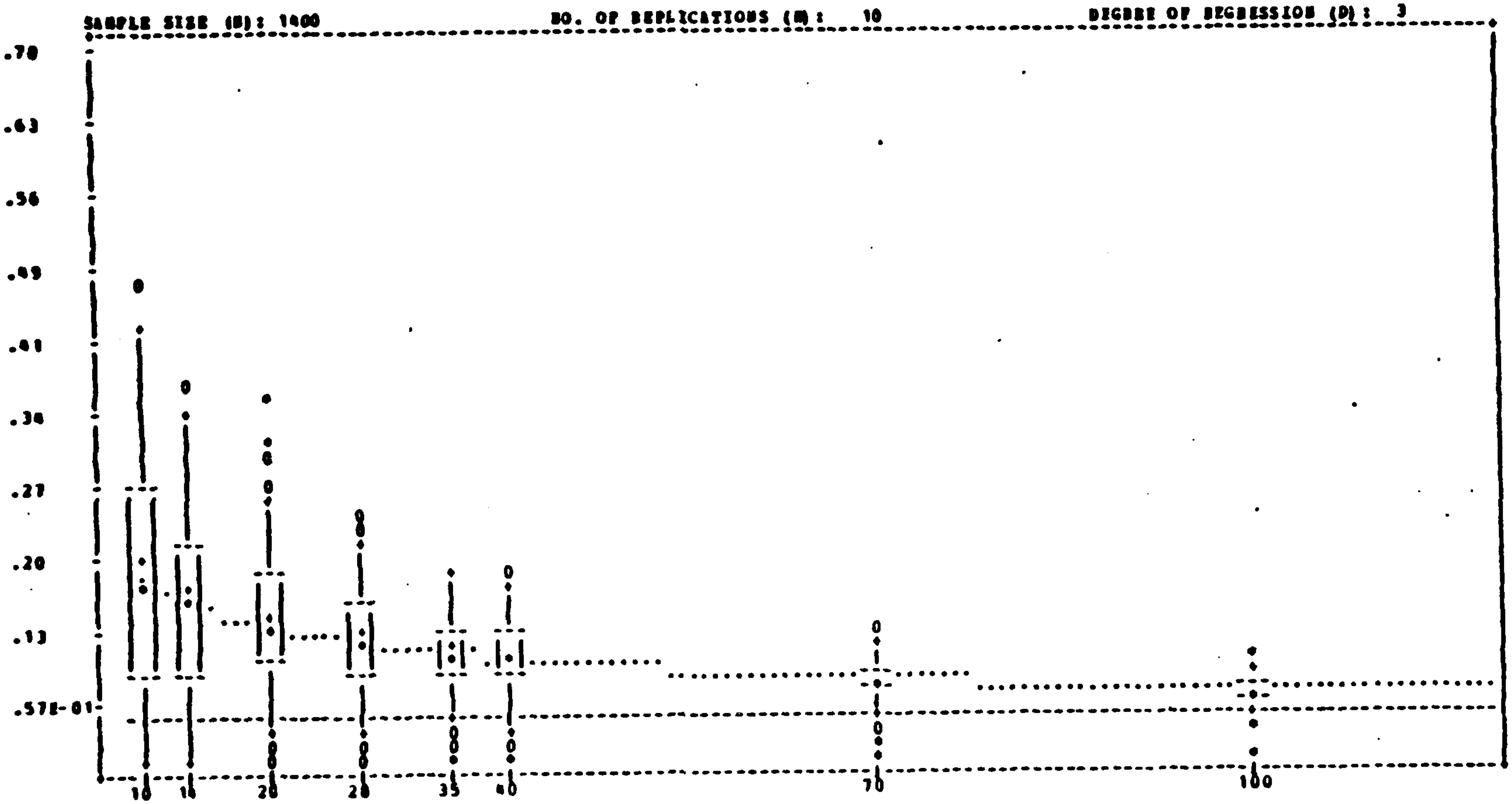
SUBSAMPLE SIZE	10	14	20	28	35	40	70	100
MEAN	.2893	.2261	.1835	.1522	.1326	.1242	.9232E-01	.7634E-01
STD	.1276	.8635E-01	.5907E-01	.3959E-01	.3159E-01	.2718E-01	.1482E-01	.1101E-01
STD MEAN	.3611E-02	.2731E-02	.2233E-02	.1788E-02	.1554E-02	.1453E-02	.1048E-02	.9305E-03
SKEWNESS	0.790	0.732	0.961	0.452	0.272	0.431	0.257	0.041
KURTOSIS	1.150	0.934	2.909	0.262	0.069	0.052	-0.112	0.095

MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:	.400858E-01	3.96488	-27.8635	122.461
VARIANCE OF REGRESSION - COEFFICIENTS:	.130551E-05	.938788E-02	9.29861	456.580
STD DEV OF REGRESSION - COEFFICIENTS:	.114259E-02	.968911E-01	3.04936	21.3677

VERTICAL SCALE: YMIN = 0.0
YMAX = 0.7000

ESTIMATOR: ESTIMATES OF THE STANDARD DEVIATION OF THE CORRELATION COEFFICIENT FOR A BIVARIATE NORMAL (0,1,0.5) JACKKNIFE

Figure 10. Estimates of the Standard Deviation of the Correlation Coefficient for a Bivariate Normal (0,1,0.5) Jackknife



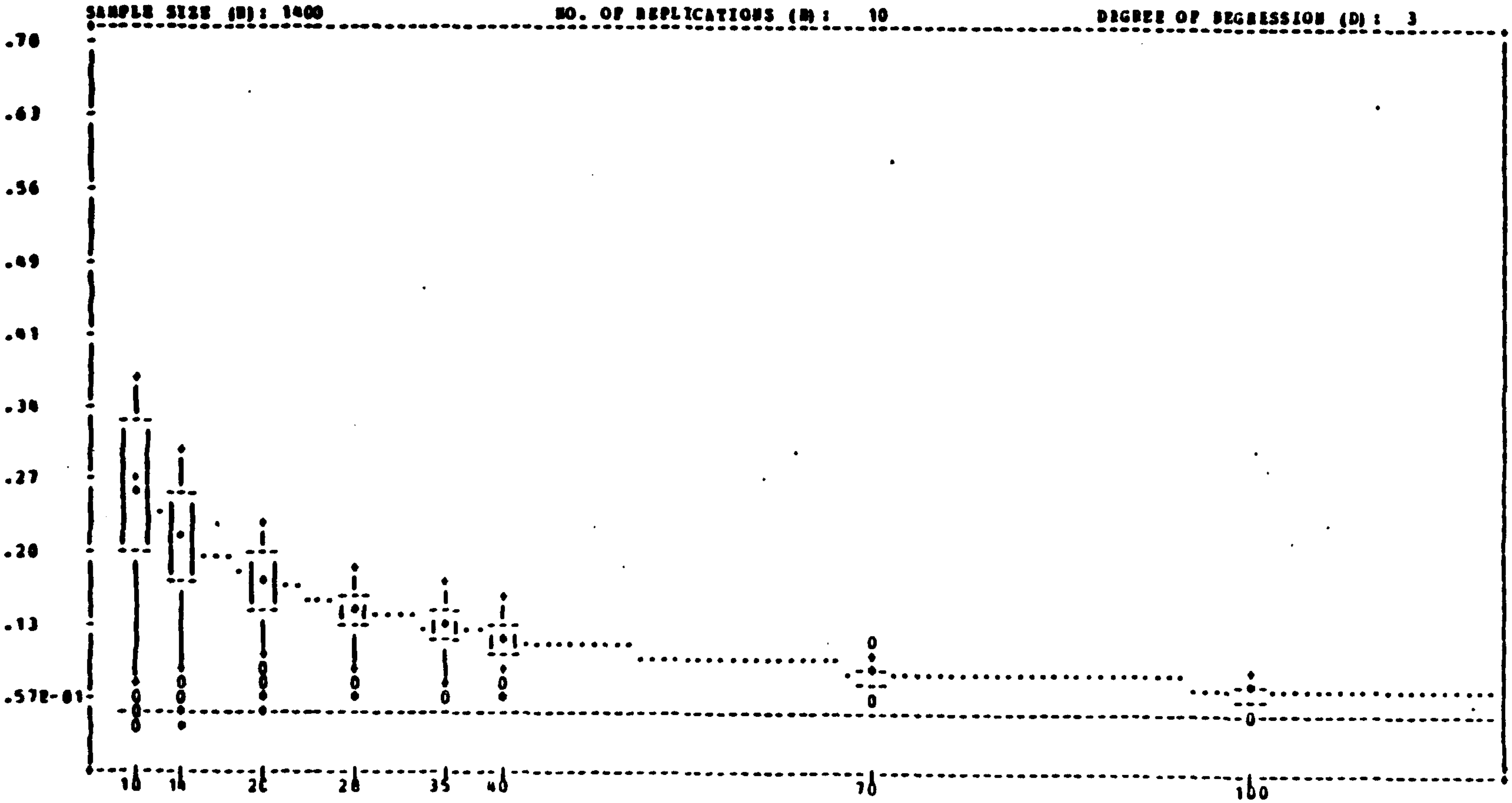
SUBSAMPLE SIZE	10	14	20	28	35	40	70	100
MEAN	.1013	.1572	.1395	.1213	.1086	.1054	.8268E-01	.7044E-01
S.D.	.1182	.9604E-01	.7361E-01	.5714E-01	.4638E-01	.3975E-01	.2434E-01	.1576E-01
STD MEAN	.3160E-02	.3037E-02	.2782E-02	.2556E-02	.2319E-02	.2125E-02	.1721E-02	.1332E-02
SKEWNESS	-0.321	-0.414	-0.620	-0.785	-0.949	-0.930	-1.492	-1.546
KURTOSIS	-0.556	-0.887	-0.327	-0.043	0.423	0.873	3.107	4.759

MEAN OF REGRESSION OF AVERAGES - COEFFICIENTS:	.425502E-01	3.19594	-32.9957	151.094
VARIANCE OF REGRESSION - COEFFICIENTS:	.206526E-05	.222224E-01	12.1487	512.978
STD DEV OF REGRESSION - COEFFICIENTS:	.143710E-02	.149072	3.48493	22.6489
REGRESSION ON VARIANCE - COEFFICIENTS:	-.369073E-01	.463486	2.17392	-5.92218

VERTICAL SCALE: YMIN = 0.0
YMAX = 0.7000

ESTIMATOR: ESTIMATES OF THE S.D. OF THE CORRELATION COEFFICIENT FOR A BIVARIATE NORMAL (0,1,0.5) DELTA METHOD (S.D.=0 IF VAR<0)

Figure 11. Estimates of the Standard Deviation of the Correlation Coefficient for a Bivariate Normal (0,1,0.5) Delta Method (S.D. = 0 if Var < 0)



SUBSAMPLE SIZE	10	14	20	28	35	40	70	100
MEAN	.2674	.2172	.1782	.1481	.1312	.1226	.9164E-01	.7035E-01
STD	.8203E-01	.5527E-01	.3706E-01	.2642E-01	.2208E-01	.1816E-01	.1392E-01	.7632E-02
STD MEAN	.2192E-02	.1746E-02	.1401E-02	.1181E-02	.1104E-02	.9705E-03	.7723E-03	.6450E-03
SKEWNESS	-0.513	-0.466	-0.437	-0.389	-0.299	-0.143	-0.220	-0.303
KURTOSIS	-0.711	-0.469	-0.205	-0.076	-0.303	-0.221	-0.293	0.174

MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:	.407671E-01	3.91095	-29.7087	113.313
VARIANCE OF REGRESSION - COEFFICIENTS:	.237218E-06	.577969E-03	.150000	6.03333
STD DEV OF REGRESSION - COEFFICIENTS:	.487051E-03	.240413E-01	.387298	2.45628
REGRESSION ON VARIANCE - COEFFICIENTS:	-.135886E-01	.285898	-1.30839	3.03682

VERTICAL SCALE: YMIN = 0.0
YMAX = 0.7000

ESTIMATOR: ESTIMATES OF THE STANDARD DEVIATION OF THE CORRELATION COEFFICIENT FOR A BIVARIATE NORMAL (0,1,0.5) NORMAL THEORY

Figure 12. Estimates of the Standard Deviation of the Correlation Coefficient for a Bivariate Normal (0,1,0.5) Normal Theory

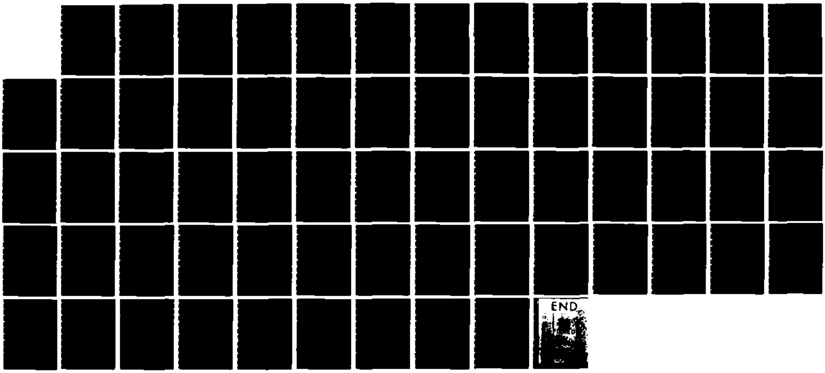
APPENDIX A

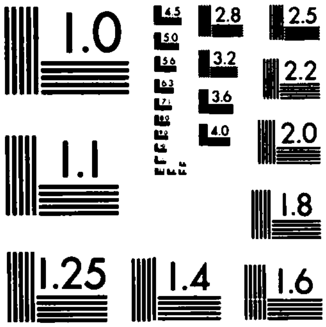
SIMTB1 (VERSION 1) PROGRAM LISTING

```
C
C                SIMTB1  PROGRAM  LISTING
C
C
C
C PURPOSE    TO GENERATE REGRESSION ADJUSTED ESTIMATES AND BOX PLOTS
C            OF ESTIMATES OF AN INPUT RAW DATA SERIES X CONTAINING M
C            (REPLICATIONS) OF N VALUES EACH.  UP TO 3 ESTIMATING
C            FUNCTIONS CAN BE USED.  THE GRAPHS CAN ALL BE OF THE SAME
C            SCALE OR SCALED INDIVIDUALLY.
C
C
C DESCRIPTION OF PARAMETERS
C
C    X       REAL*4 ARRAY CONTAINING DATA.
C            A MAXIMUM OF 50,000 DATA ELEMENTS CAN BE STORED IN X.
C
C    N       NUMBER OF DATA ELEMENTS PER SECTION (N IS SAMPLE SIZE).
C            N CANNOT EXCEED 50,000 AND M*N MUST NOT EXCEED 50,000.
C
C    M       NUMBER OF SECTIONS (REPLICATIONS).
C            M CANNOT EXCEED 100 AND M*N MUST NOT EXCEED 50,000.
C
C    NE      INTEGER ARRAY OF SIZE 8 CONTAINING SUBSAMPLE SIZES FOR N.
C            THE VALUES OF NE MUST BE FROM SMALLEST TO LARGEST.
C            NO ELEMENT OF THE ARRAY NE CAN BE GREATER THAN N.
C            M*(N/NE(1)) MUST NOT EXCEED 12,500.
C
C    L       NUMBER OF SUBSAMPLE SIZES FROM NE(8) THAT WILL BE USED TO
C            SECTION N.
C            IT IS ALSO THE NUMBER OF BOXPLOTS THAT WILL BE PRODUCED.
C
C    D       DEGREE OF REGRESSION FOR MEAN AND VARIANCE REGRESSIONS.
C            D WILL BE REDUCED BY RAGE IF THE SAMPLE IS NOT LARGE
C            ENOUGH. D MUST BE 1,2 OR 3. D=0 WILL IGNORE REGRESSIONS.
C
C            *** SCALING ***
C              SCALING IS ACCOMPLISHED BY TAKING THE SMALLEST AND THE
C              LARGEST ESTIMATE VALUES FROM ALL ESTIMATING FUNCTIONS
C              AND FROM FOR ALL SUBSAMPLE SIZES.
C                 THE SEI PARAMETER ALLOWS THE USER TO SCALE THE GRAPHS
C              OF EACH ESTIMATOR INDIVIDUALLY OR TO SCALE THEM ALL TO
C              THE SAME SCALE.  SCALING ALL TO THE SAME SCALE IS
C              ACCOMPLISHED BY TAKING THE MINIMUM AND MAXIMUM  ESTIMATE
C              FROM ALL THE ESTIMATORS USING NE(1) SUBSAMPLE SIZE.
C                 THE RG PARAMETER ALLOWS THE USER TO RREDUCE THE VER-
C              TICAL SCALE TO: THE UPPER QUARTILE DISTANCE + 1.5 TIMES
C              INTERQUARTILE DISTANCE AS THE MAX VALUE AND THE LOWER
```

AD-A136 812 SIMTBED: A GRAPHICAL TEST BED FOR ANALYZING AND REPORTING THE RESULTS OF A STATISTICAL SIMULATION EXPERIMENT(U) NAVAL POSTGRADUATE SCHOOL MONTEREY CA 2/2

UNCLASSIFIED H W DRUEG SEP 83 F/G 9/2 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

```
C               QUARTILE - 1.5 TIMES THE INTERQUARTILE DISTANCE AS THE
C               MIN VALUE.  THE INTERQUARTILE DISTANCE IS COMPUTED FROM
C               THE SAMPLE OF ESTIMATES FROM THE NE(1) SUBSAMPLE SIZE.
C               IF THERE ARE NO ESTIMATES OUTSIDE THESE MIN AND MAX
C               VALUES THEN THE SCALE IS TO THE FIRST VALUE WITHIN.
C               IF THERE ARE ESTIMATES OUTSIDE THESE LIMITS THEN THEY
C               ARE COUNTED AND THE NUMBER PRINTED AT THE ENDS OF THE
C               BOX PLOTS.
C                  THE SVS PARAMET ALLOWS THE USER TO SET THE VERTICAL
C               SCALE. WHEN THE VERTICAL SCALE IS SET THE SEI PARAMETER
C               IS IGNORED AND THE VERTICAL SCALE BECOMES YMIN AND YMAX.
C
C     RG        RG=0 DO NOT REDUCE THE VERTICAL SCALE OF THE GRAPHS.
C               RG=1 REDUCE GRAPHICS VERTICAL SCALE TO UPPER (LOWER)
C                    QUARTILE + (-) INTERQUARTILE DISTANCE.
C
C     SEI       SEI=0 DO NOT SCALE ESTIMATORS' GRAPHS INDIVIDUALLY.
C               SEI=1 SCALE ESTIMATORS' GRAPHS INDIVIDUALLY.
C
C     SVS       SVS=0  PROGRAM WILL CALCULATE VERTICAL SCALE.
C               SVS=1  USER SETS VERTICAL SCALE TO YMIN AND YMAX.
C
C     YMIN      LOW VALUE OF VERTICAL SCALE. SET BY USER WHEN SVS=1
C
C     YMAX      HIGH VALUE OF VERTICAL SCALE. SET BY USER WHEN SVS=1
C
C     NEST      NUMBER OF ESTIMATORS THAT WILL BE USED TO CALCULATE
C               STATISTICAL PARAMETER FROM X DATA.
C               NEST MUST BE 1,2 OR 3.
C
C     EST1      NAMES OF THE ESTIMATOR FUNCTIONS THAT WILL BE USED TO
C     EST2      CALCULATE THE STATISTICAL PARAMETER.
C     EST3      CALL SEQUENCE ON EACH FUNCTION IS: CALL FNAME(X,N) WHERE
C               X IS THE DATA ARRAY AND N IS THE NUMBER OF DATA POINTS.
C               THEY MUST BE DECLARED IN THE CALLING PROGRAM (RAGE) IN
C               THE ORDER THEY ARE USED. DUMMY VARIABLES MUST BE INSERTED
C               WHEN THERE ARE LESS THAN 3 ESTIMATORS.
C
C     TTL1      TITLES ASSOCIATED WITH EACH ESTIMATOR (EST1,2,3).  A MAX
C     TTL2      OF 120 CHARACTERS CAN BE USED TO DESCRIBE EACH ESTIMATOR.
C     TTL3      EACH TITLE MUST BE DECLARED AS REAL*8(15) ARRAYS UNLESS
C               PASSED AS AN ARGUMENT OF THE CALLING PROGRAM RAGE.
C               WHEN PASSING THE TITLE AS AN ARGUMENT THERE  MUST BE A
C               MINIMUM OF 120 CHARACTERS BETWEEN APOSTROPHES.
C
C*****************************************************************************
C
```

```
      SUBROUTINE SIMTB1(X,N,M,NE,L,D,RG,SEI,SVS,YMIN,YMAX,NEST,EST1,
     +TTL1,EST2,TTL2,EST3,TTL3)
      REAL X(50000),ULH(4),Y(12500)
      REAL*8 TTL1(15),TTL2(15),TTL3(15)
      INTEGER NE(8),RG,SEI,SVS,SM
      INTEGER D,I,NEST,TEST
C
      SM=NE(1)
      MN = M*N
      LT=L-1
      IF(LT.EQ.0) GO TO 13
      DO 11 I=1,LT
         I1=I+1
         IF(NE(I).GT.NE(I1)) WRITE(6,110)
   11 CONTINUE
   13 TEST=0
      IF(NEST.EQ.1.OR.NEST.EQ.2.OR.NEST.EQ.3)  GO TO 1
      WRITE(6,106)
      TEST=1
    1 IF(MN.LE.50000) GO TO 2
      WRITE(6,105)
      TEST=1
    2 IF(M.GE.1.AND.M.LE.100) GO TO 3
      WRITE(6,104)
      TEST=1
    3 IF(L.GE.1.AND.L.LE.8) GO TO 4
      WRITE(6,103)
      TEST=1
    4 IF(D.LE.3) GO TO 5
      WRITE(6,108)
      TEST=1
    5 K=N/NE(L)
      IF(K.GE.1) GO TO 6
      WRITE(6,107)
      TEST=1
    6 K=M*(N/NE(1))
      IF(K.LE.12500) GO TO 7
      WRITE(6,109)
      TEST=1
    7 IF (TEST.NE.0) GO TO 80
      ULH(2)=YMIN
      ULH(4)=YMAX
C     DETERMINE HOW EACH GRAPH IS TO BE SCALED.
      IF (SVS .EQ. 1) GO TO 50
      IF (SEI .EQ. 1) GO TO 75
C     ************************************************************
C     *GRAPH ALL ESTIMATORS TO THE SAME SCALE OF ESTIMATOR W/WIDEST PTS.
C     ************************************************************
```

```
      ULH(2)=1.E30
      ULH(4)=-1.E30
      DO 10 IK=1,I
C       FIND VERTICAL SCALE FOR 1ST ESTIMATOR.
        CALL SECEST(X,N,M,NE(IK),EST1,Y,KP)
        IF(RG.EQ.1) CALL DELETO(Y,KP,YMAX,YMIN)
        IF(RG.NE.1) CALL MAXMIN(Y,KP,YMAX,YMIN)
        IF(YMIN .LT. ULH(2)) ULH(2)=YMIN
        IF(YMAX .GT. ULH(4)) ULH(4)=YMAX
C
C       FIND VERTICAL SCALE FOR 2ND ESTIMATOR.  KEEP WIDEST PAIR
        IF(NEST .LT. 2) GO TO 10
        CALL SECEST(X,N,M,NE(IK),EST2 ,Y,KP)
        IF(RG.EQ.1) CALL DELETO(Y,KP,YMAX,YMIN)
        IF(RG.NE.1) CALL MAXMIN(Y,KP,YMAX,YMIN)
        IF(YMIN .LT. ULH(2)) ULH(2)=YMIN
        IF(YMAX .GT. ULH(4)) ULH(4)=YMAX
C
C       FIND VERTICAL SCALE FOR 3RD ESTIMATOR.  KEEP WIDEST PAIR****
        IF(NEST .LT. 3) GO TO 10
        CALL SECEST(X,N,M,NE(IK),EST3 ,Y,KP)
        IF(RG.EQ.1) CALL DELETO(Y,KP,YMAX,YMIN)
        IF(RG.NE.1) CALL MAXMIN(Y,KP,YMAX,YMIN)
        IF(YMIN .LT. ULH(2)) ULH(2)=YMIN
        IF(YMAX .GT. ULH(4)) ULH(4)=YMAX
   10 CONTINUE
C     RETURN CALCULATED SCALE TO CALLER
      YMIN=ULH(2)
      YMAX=ULH(4)
C
C     PROCESS BOXPLOTS USING FIXED VERTICAL SCALE IN VECTOR ULH
C     ONE CALL FOR EACH ESTIMATOR USED.
   50 CALL PRST(X,N,M,EST1,NE,L,RG,D,ULH,Y)
      WRITE(6,101) ULH(2),ULH(4)
      WRITE(6,102) TTL1
      IF (NEST. LT. 2) GO TO 80
      CALL PRST(X,N,M,EST2,NE,L,RG,D,ULH,Y)
      WRITE(6,101) ULH(2),ULH(4)
      WRITE(6,102) TTL2
      IF (NEST. LT. 3) GO TO 80
      CALL PRST(X,N,M,EST3,NE,L,RG,D,ULH,Y)
      WRITE(6,101) ULH(2),ULH(4)
      WRITE(6,102) TTL3
      GO TO 80
C     ******************************************************
C     *GRAPH EACH ESTIMATOR SCALED TO ITS WIDEST POINTS.*
C     ******************************************************
C     FIND VERTICAL SCALE FOR 1ST ESTIMATOR AND GRAPH.
```

```
   75 CALL SECEST(X,N,M,NE(1),EST1,Y,KP)
      IF(RG.EQ.1) CALL DELETO(Y,KP,YMAX,YMIN)
      IF(RG.NE.1) CALL MAXMIN(Y,KP,YMAX,YMIN)
      ULH(2)=YMIN
      ULH(4)=YMAX
      CALL PRST(X,N,M,EST1,NE,L,RG,D,ULH,Y)
      WRITE(6,101) ULH(2),ULH(4)
      WRITE(6,102) TTL1
      IF(NEST.LT. 2) GO TO 80
C
C     FIND VERTICAL SCALE FOR 2ND ESTIMATOR AND GRAPH.
      CALL SECEST(X,N,M,NE(1),EST2 ,Y,KP)
      IF(RG.EQ.1) CALL DELETO(Y,KP,YMAX,YMIN)
      IF(RG.NE.1) CALL MAXMIN(Y,KP,YMAX,YMIN)
      ULH(2)=YMIN
      ULH(4)=YMAX
      CALL PRST(X,N,M,EST2,NE,L,RG,D,ULH,Y)
      WRITE(6,101) ULH(2),ULH(4)
      WRITE(6,102) TTL2
      IF(NEST.LT. 3) GO TO 80
C
C     FIND VERTICAL SCALE FOR 3RD ESTIMATOR AND GRAPH.
      IF (SVS.EQ.1) GO TO 78
      CALL SECEST(X,N,M,NE(1),EST3 ,Y,KP)
      IF(RG.EQ.1) CALL DELETO(Y,KP,YMAX,YMIN)
      IF(RG.NE.1) CALL MAXMIN(Y,KP,YMAX,YMIN)
      ULH(2)=YMIN
      ULH(4)=YMAX
   78 CALL PRST(X,N,M,EST3,NE,L,RG,D,ULH,Y)
      WRITE(6,101) ULH(2),ULH(4)
      WRITE(6,102) TTL3
   80 CONTINUE
C
  102 FORMAT(1X,'ESTIMATOR: ',15A8)
  101 FORMAT(1X,'VERTICAL SCALE:  YMIN =',F10.4,/,16X,'YMAX =',F10.4/)
  103 FORMAT(' *** ERROR...L MUST BE AN INTEGER BETWEEN 1 AND 8. ***')
  104 FORMAT(' *** ERROR...M MUST BE AN INTEGER BETWEEN 1 AND 100. ***')
  105 FORMAT(' *** ERROR...M*N EXCEEDS 50,000. ***')
  106 FORMAT(' *** ERROR...NEST MUST BE 3 OR LESS. ***')
  107 FORMAT(' *** ERROR...N/NE(L) MUST BE 1 OR GREATER TO COMPUTE',
     +' STATISTICS.',/,'   INCREASE N OR DECREASE NE(L). ***')
  108 FORMAT(' *** ERROR...D MUST BE LESS THAN OR EQUAL TO 3. ***')
  109 FORMAT(' *** ERROR...M*(N/NE(1)) MUST NOT EXCEED 12,500. ***')
  110 FORMAT(' *** WARNING... NE ARRAY ELLEMENTS ARE NOT IN ORDER OF '/
     +'      INCREASING SIZE. IF NE(1) IS NOT SMALLEST ELEMENT, SCALING'/
     +'      MAY CAUSE POINTS TO FALL OUTSIDE RANGE OF SCALE.')
      RETURN
      END
```

```
C
C*********************************************************************
C
      SUBROUTINE ERST(X,N,M,EST,NE,L,RG,UD,ULH,Y)
C     REGRESSION ADJUSTED ESTIMATE
C     CALCULATES ESTIMATES FROM USER DATA USING "EST" FUNCTION
C     PLOTS BASIC OR RETRENCHED GRAPH ON LINE PRINTER
C
C     EST = NAME OF USER WRITTEN ESTIMATING FUNCTION.
C           USAGE:     FUNCTION_NAME(X,N)
C           WHERE X IS A VECTOR WITH N ENTRIES
C
C     M = NO. OF REPLICATIONS (MUST BE <= 100)
C     N = NUMBER OF VALUES IN EACH REPLICATION (M*N MUST BE <= 50000)
C     X = USERS VECTOR WITH M CONSECUTIVE BATCHES OF N VALUES EACH
C     L = NO. OF SECTION SIZES (MUST BE BETWEEN 1 AND 8)
C     NE = ARRAY WITH THE L SUBSAMPLE SIZES (MUST BE IN ASCENDING ORDER)
C     UD = DEGREE OF THE REGRESSION (MUST BE <= 3 & <= L-1 )
C     ULH= XMIN,YMIN,XMAX,YMAX  IN USER UNITS
C          ONLY ULH(2) AND ULH(4) NEED TO BE PASSED. OTHERS CALC. HERE
C
      DIMENSION   NE(L)
      INTEGER NB(8),LOCX(8),D1,IWIDTH,UD,D,LT,DT,RG,RNEK
      INTEGER*2 PLOT(122,50), CBAR,BLK,DASH,CSTR,NUM(10),DOT
      REAL*4 X(50000),ULH(4),DLH(4),Y(12500)
      REAL*4 RH(8,100),STAT(8,6),VT(8)
      REAL*8 SUM2,SUM3,SUM4,LABEL(5)
      REAL*4 RA(8,4),RV(8,4),B(4,100),V(4),BA(4),BV(4),BS(4),RT(8),ET(8)
      DATA DLH/1.,1.,122.,50./
      DATA BLK/' '/, DASH/'-'/, CBAR/'|'/, DOT/'.'/
      DATA LABEL/'MEAN','STD','STD MEAN','SKEWNESS','KURTOSIS'/
      D=MINO(3,UD,L-1)
      IX1=8
      IX2=4
      MN=M*N
      D1=D+1
      IWIDTH= IFIX(DLH(3))
C     BUILD REGRESSION MATRICES FOR AVERAGES AND VARIANCES
      DO 84  K=1,L
       DO 86 J=1,D1
        T=FLOAT(NE(L))/FLOAT(NE(K))
        RA(K,J)=T**(J-1)
        RV(K,J)=T**((FLOAT(J))/2.0)
   86  CONTINUE
   84 CONTINUE
C     CLEAR PLOT ARRAY
      DO 3 J=1,50
       DO 4 I=1,122
```

```
          PLOT(I,J)=BLK
    4   CONTINUE
    3 CONTINUE
C     SET HORIZONTAL XMIN, XMAX
      ULH(1)=.7*NE(1)
      ULH(3)=1.2*NE(L)
C     SET SCALE
      CALL SCALE(ULH,DLH)
C     COMPUTE LOCATION OF BOXPLOTS ALONG X-AXIS
      LAST=-1
      DO 5 K=1,L
       NB(K)=N/NE(K)
       LOCX(K)=(NE(K)-ULH(1))*(DLH(3)-DLH(1))/(ULH(3)-ULH(1))+DLH(1)+.5
       IF(LOCX(K).LT. LAST+4) LOCX(K)=LAST+4
       LAST=LOCX(K)
    5 CONTINUE
C
      DO 80 K=1,L
       NBK=NB(K)
       RNEK=NE(K)
C      SECTION 6 COMPUTE ESTIMATORS FOR SIZE K
       CALL SECEST(X,N,M,RNEK,EST,Y,KP)
C      AVERAGE ESTIMATES OF SIZE NE(K) FOR EACH OF M REPLICATIONS
       KP=0
       DO 10 I=1,M
        RH(K,I)=0.
        DO 15 J=1,NBK
         KP=KP+1
         RH(K,I)=RH(K,I)+Y(KP)
   15   CONTINUE
        RH(K,I)=RH(K,I)/FLOAT(NBK)
   10  CONTINUE
      CALL BOXPRT(Y,KP,LOCX(K),PLOT,RG)
      IF (K .GT. 8) GO TO 80
C     COMPUTE MEAN AND MOMENT ESTIMATES
      XMEAN=0.
      DO 180 IM1=1,KP
       XMEAN=XMEAN+Y(IM1)
  180 CONTINUE
      XMEAN=XMEAN/FLOAT(KP)
      SUM2 = 0.0D0
      SUM3 = 0.0D0
      SUM4 = 0.0D0
      DO 190 IP1=1,KP
      DEV = Y(IP1) - XMEAN
      SUM2 = SUM2 + DEV * DEV
      SUM3 = SUM3 + DEV ** 3
      SUM4 = SUM4 + DEV ** 4
```

```
  190 CONTINUE
C
C     CHECK TO INSURE SAMPLE SIZE IS LARGE ENOUGH FOR
C     EACH MOMENT COMPUTATION.
      IF (KP.LT.2) GO TO 7
      VAR = SUM2 / (KP - 1.0)
      STDV = SQRT(VAR)
    7 IF (KP.LT.3) GO TO 8
      XSUM3 = SNGL(SUM3) * KP /( (KP-1.) * (KP-2.))
      SKEW = XSUM3 / STDV ** 3
    8 IF (KP.LT.4) GO TO 9
      XSUM4 = SNGL(SUM4)*((KP-2.)*KP+3.)/((KP-1.)*(KP-2.)*(KP-3.))
      XSUM4 = XSUM4 - VAR*VAR*3.*(KP-1.)*(KP+KP-3.)/(KP*(KP-2.)*(KP-3))
      CKURT = XSUM4 / (VAR * VAR) - 3.
    9 STAT(K,1)=XMEAN
      STAT(K,2)=STDV
      STAT(K,3)=STDV/SQRT(FLOAT(KP))
      STAT(K,4)=SKEW
      STAT(K,5)=CKURT
      STAT(K,6)=VAR
   80 CONTINUE
C
C     IF D1.LT.2 THEN NO REGRESSIONS OR PLOTING CAN BE DONE
C
      IF(D1.LT.2) GO TO 113
      DO 92 K=1,M
         DO 47 J=1,L
            RT(J)=RH(J,K)
  47  CONTINUE
      CALL RREG(RA,RT,BT,L,D1,IX1,IX2)
      B(1,K)=BT(1)
      DO 23 KT=2,D1
         B(KT,K)=BT(KT)*NE(L)**(KT-1)
   23 CONTINUE
   92 CONTINUE
C
C     AVERAGE REGRESSION COEFF. OVER M REPLICATIONS & CALC. VARIANCE
C
      DO 94 I=1,D1
       BA(I)=0.
       BV(I)=0.
       DO 95 J=1,M
        BA(I)=BA(I)+B(I,J)
        BV(I)=BV(I)+B(I,J)**2
   95  CONTINUE
       BA(I)=BA(I)/FLOAT(M)
       IF(M.EQ.1)GO TO 94
       BV(I)=(BV(I)-M*BA(I)**2)/(M*(M-1.))
```

```
      BS(I)=BV(I)**.5
   94 CCNTINUE
C
C     ESTABLISH BEGRESSION LINE & ASYMPTOTE
C
      DC 98 I=3,IWIDTH
C      MAP I FROM DEVICE SPACE TO USER SPACE
       UX=(I-DLH(1))*(ULH(3)-ULH(1))/(DLH(3)-DLH(1)) + ULH(1)
C      COMPUTE THE Y VALUE FROM X AND THE REGRESSION COEFFICIENTS.
       UY=BA(1)
       DC 99 J=1,L
        UY=UY+BA(J+1)/UX**J
   99  CONTINUE
C
C      MAP THE Y VALUE FROM USER SPACE TO DEVICE SPACE
       J=(UY-ULH(2))*(DLH(4)-DLH(2))/(ULH(4)-ULH(2)) + DLH(2) + .5
       IF(J .LT. 1 .OR. J .GT. 50) GO TO 98
       IF(PLOT(I,J) .EQ. BLK) PLOT(I,J)=DOT
   98 CCNTINUE
C
C     SCALE ASYMPTOTE, BETA0, AND PLOT ACROSS PLCT.
      J=(BA(1)-ULH(2))*(DLH(4)-DLH(2))/(ULH(4)-ULH(2)) + DLH(2) + .5
      IF(J .LT. 1 .OR. J .GT. 50) GO TO 117
      DC 120 I=3,IWIDTH
       IF(PLOT(I,J) .EQ. BLK) PLOT(I,J)=DASH
  120 CCNTINUE
C
C     REGRESSION CN VARIANCES FROM EACH SEGMENT WITH A VARIANCE.
C
  117 K=M*(N/NE(L))
      LT=L
      DT=D1
      DC 111 I=1,L
         IF(K.GE.2) GO TO 112
         LT=LT-1
         K=M*(N/NE(LT))
  111 CCNTINUE
  112 IF(LT.LT.DT) DT=LT
      IF(DT.LT.2) GO TO 113
      DC 48 J=1,L
         VT(J)=STAT(J,6)*(NE(J)**0.5)
   48 CCNTINUE
      CALL RBEG(BV,VT,V,LT,DT,IX1,IX2)
      DC 77 I=1,LT
         V(I) = V(I)*NE(L)**(FLOAT(I)/2.)
   77 CCNTINUE
C
C     PLOT
```

```
  113 WRITE(6,102)
      WRITE(6,161) N,M,D
      WRITE(6,101)
      DO 90 J=1,50
       K=51-J
       IF(MOD(K,5).NE.0) GO TO 85
       YLABEL=(K-DLH(2))*(ULH(4)-ULH(2))/(DLH(4)-DLH(2)) + ULH(2)
       WRITE(6,103)  YLABEL,(PLOT(I,K),I=1,IWIDTH)
       GO TO 90
   85  CONTINUE
       WRITE(6,100)  (PLOT(I,K),I=1,IWIDTH)
   90 CONTINUE
C     LABEL X-AXIS BY REUSING PLOT MATRIX.
      DO 115 I=1,122
        PLOT(I,1)=DASH
        PLOT(I,2)=BLK
  115 CONTINUE
      DO 130 J=1,I
       PLOT(LOCX(J),1)=CBAR
       IK = NE(J)
       IX = LOCX(J)
       CALL NUMPRT(IX,2,IK,PLOT)
  130 CONTINUE
      WRITE(6,106) (PLOT(I,1),I=1,IWIDTH)
      WRITE(6,104) (PLOT(I,2),I=1,IWIDTH)
      L8=L
      IF(L8 .GT. 8) L8=8
      WRITE(6,156)
      WRITE(6,146) (NE(I),I=1,L8)
      WRITE(6,157) LABEL(1), (STAT(K,1),K=1,L8)
C     CHECK TO INSURE SAMPLE SIZE IS LARGE ENOUGH TO COMPUTE
C     STATISTICS BEFORE ATTEMPTING TO PRINT STATS.
      LT=L8
      L1=L8
      DO 21 I=1,LT
         K1 = M*(N/NE(L1))
         IF (K1.GE.2) GO TO 11
         L1 = L1-1
   21 CONTINUE
      GO TO 14
   11 WRITE(6,157) LABEL(2), (STAT(K,2),K=1,L1)
      WRITE(6,157) LABEL(3), (STAT(K,3),K=1,L1)
      LT = L1
      DO 22 I=1,LT
         K1 = M*(N/NE(L1))
         IF (K1.GE.3) GO TO 12
         L1 = L1-1
   22 CONTINUE
```

```
      GO TO 14
   12 WRITE(6,158) LABEL(4), (STAT(K,4),K=1,L1)
      LT = L1
      DO 33 I=1,LT
         K1 = M*(N/NE(L1))
         IF (K1.GE.4) GO TO 13
         L1 = L1-1
   33 CONTINUE
      GO TO 14
   13 WRITE(6,158) LABEL(5), (STAT(K,5),K=1,L1)
      IF(D1.LT.2) GO TO 444
      WRITE(6,151) (BA(I),I=1,D1)
   14 IF(M.LT.2)GO TO 444
      WRITE(6,152) (BV(I),I=1,D1)
      WRITE(6,153) (BS(I),I=1,D1)
  444 IF(DT.LE.1) GO TO 999
      WRITE(6,159) (V(I),I=1,DT)
  999 WRITE(6,162)
  100 FORMAT(9X,'|',122A1,'|')
  101 FORMAT(9X,'+',122('-'),'+')
  102 FORMAT('1')
  103 FORMAT(' ',G8.2,'-',122A1,'|')
  104 FORMAT(10X,122A1)
  106 FORMAT(9X,'+',122A1,'+')
  107 FORMAT(10G13.8)
  151 FORMAT(/,' MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:',4G20.6)
  152 FORMAT(' VARIANCE OF REGRESSION - COEFFICIENTS:',8X,4G20.6)
  153 FORMAT(' STD DEV OF REGRESSION - COEFFICIENTS:',9X,4G20.6)
  156 FORMAT(/' SUBSAMPLE')
  146 FORMAT(' SIZE     ',8(I8,6X),/)
  157 FORMAT(1X,A8,8G14.4)
  158 FORMAT(1X,A8,8F14.3)
  159 FORMAT(/,' REGRESSION ON VARIANCE - COEFFICIENTS: ',7X,4G20.6)
  161 FORMAT(9X,'SAMPLE SIZE (N):',I5,20X,'NO. OF REPLICATIONS (M):',
     +I5,18X,'DEGREE OF REGRESSION (D):',I3)
  162 FORMAT(/)
      RETURN
      END
C
C*****************************************************************************
C
      SUBROUTINE BOXPRT(Y,NY,IX,PLOT,RG)
C     PREPARES BOXPLOT FROM VECTOR Y (IN 2-D ARRAY PLOT)
      REAL Y(NY),ULH(4),DLH(4)
      INTEGER RG
      INTEGER*2 PLOT(122,50),DASH,CBAR,CROSS,CSTR,CC,NUM(10)
      LOGICAL*1 IFLAG
      DATA DASH/'-'/,CBAR/'|'/,CSTR/'*'/,CROSS/'+'/,CO/'0'/
```

```
      IF(NY .GE. 9) GO TO 5
C     WHEN LESS THAN 9 POINTS JUST SHOW THE POINTS
      DO 8 I=1,NY
       J=(Y(I)-YMIN)*VSCALE + 1.
C      IGNORE VALUE IF IT FALLS OUTSIDE WINDOW
       IF(J.GT.50  .OR.  J.LT.1)  GO TO 8
       PLOT(IX,J)=CO
    8 CONTINUE
      SUM=0.
      DO 88 I=1,NY
       SUM=SUM+Y(I)
   88 CONTINUE
      SUM=SUM/FLOAT(NY)
      MEAN=(SUM-YMIN)*VSCALE+1
      PLOT(IX,MEAN)=CSTR
      GO TO 99
    5 CONTINUE
      LFLAG=.FALSE.
      P25 =PCTL(Y,NY,.25)
      P75 =PCTL(Y,NY,.75)
      P50 =PCTL(Y,NY,.50)
      IQ1=(P25-YMIN)*VSCALE+1.
      IQ2=(P50-YMIN)*VSCALE+1.
      IQ3=(P75-YMIN)*VSCALE+1.
      XLOW=2*P25-P75
      ILOW=(XLOW-YMIN)*VSCALE+1.
      XHI=2*P75-P25
      IHI=(XHI-YMIN)*VSCALE+1.
      CLOW=2.5*P25-1.5*P75
      CHI=2.5*P75-1.5*P25
C     DRAW BOX
      DO 20 I=IQ1,IQ3
       PLOT(IX-1,I)=CBAR
       PLOT(IX+1,I)=CBAR
   20 CONTINUE
      PLOT(IX-1,IQ1)=DASH
      PLOT(IX+1,IQ1)=DASH
      PLOT(IX-1,IQ3)=DASH
      PLOT(IX+1,IQ3)=DASH
C     DETERMINE IF OUTLIERS ARE TO BE COUNTED AND THE NUMBER PRINTED.
      IF (RG.EQ.1) GO TO 55
      DO 30 I=1,NY
       J=(Y(I)-YMIN)*VSCALE+1.
C      IGNORE VALUE IF IT FALLS OUTSIDE WINDOW
       IF(J.GT.50  .OR.  J.LT.1)  GO TO 30
       IF(Y(I).LT.CLOW)  PLOT(IX,J)=CSTR
       IF(Y(I).LT.CLOW)  GO TO 30
       IF(Y(I).GE.CLOW  .AND.Y(I).LT.XLOW)  PLOT(I   J)=CO
```

```
      IF(LFLAG .OR. Y(I).LT.XLOW) GO TO 25
C     THIS IS THE LOW-CROSS POINTER (1ST POINT GE XLOW)
      LFLAG=.TRUE.
      ILX=J
C     NEXT LINE ENDS UP WITH HI-CROSS POINTER (LAST POINT LE XHI)
   25 IF(Y(I).LE.XHI) IHX=J
      IF(Y(I).GT.XHI .AND.Y(I).LE.CHI) PLOT(IX,J)=CO
      IF(Y(I).GT.CHI) PLOT(IX,J)=CSTR
   30 CONTINUE
      GO TO 56
C
C     SCALE TO INTERQUARTILE +(-) INTERQUARTILE DISTANCE.
C
   55 II=0
      III=0
      DO 31 I=1,NY
       J=(Y(I)-YMIN)*VSCALE+1.
       IF(Y(I).LT.CLOW) II = II+1
       IF(Y(I).GT.CHI) III=III+1
       IF(J.GT.50 .OR. J.LT.1) GO TO 31
       IF(Y(I).GE.CLOW .AND.Y(I).LT.XLOW) PLOT(IX,J)=CC
       IF(LFLAG .OR. Y(I).LT.XLOW) GO TO 26
C      THIS IS THE LOW-CROSS POINTER (1ST POINT GE XLOW)
       LFLAG=.TRUE.
       ILX=J
C      NEXT LINE ENDS UP WITH HI-CROSS POINTER (LAST POINT LE XHI)
   26  IF(Y(I).LE.XHI) IHX=J
       IF(Y(I).GT.XHI .AND.Y(I).LE.CHI) PLOT(IX,J)=CO
   31 CONTINUE
C
C     PRINT NUMBER OF OUTLIERS UNLESS 0.
      DO 22 K=1,2
      IK=II
      J=(CLOW-YMIN)*VSCALE + 1
      IF(J.LT.0) J=1
      IF(K.EQ.2) IK=III
      IF(K.EQ.2) J=(CHI-YMIN)*VSCALE + 1
      IF(J.GT.50)J=50
      IF (IK.EQ.0) GO TO 22
      CALL NUMPRT(IX,J,IK,PLOT)
   22 CONTINUE
C
C     FILL BARS ABOVE AND BELOW THE BOX
   56 DO 32 I=ILX,IQ1
       PLOT(IX,I)=CBAR
   32 CONTINUE
      DO 33 I=IQ3,IHX
       PLOT(IX,I)=CBAR
```

```
   33 CCNTINUE
      PLOT(IX,ILX)=CROSS
      PLOT(IX,IHX)=CROSS
      PLOT(IX,IQ1)=DASH
      PLOT(IX,IQ2)=CROSS
      PLOT(IX,IQ3)=DASH
      SUM=0.
      DC 40 I=1,NY
       SUM=SUM+Y(I)
   40 CCNTINUE
      SUM=SUM/FLOAT(NY)
      MEAN=(SUM-YMIN)*VSCALE+1
      PLOT(IX,MEAN)=CSTR
   99 CCNTINUE
      RETURN
C
C*****************************************************************
C
      ENTRY SCALE(ULH,DLH)
C     CCMPUTES X,Y SCALE AND LIMITS
      XMIN=ULH(1)
      XMAX=ULH(3)
      YMIN=ULH(2)
      YMAX=ULH(4)
      HSCALE=(DLH(3)-DLH(1))/(ULH(3)-ULH(1))
      VSCALE=(DLH(4)-DLH(2))/(ULH(4)-ULH(2))
      RETURN
      END
C
C*****************************************************************
C
      SUBROUTINE RREG(XS,YS,BS,M,N,IX1,IX2)
C     ROBUST REGRESSION CN Y=X*B
C     X=M BY N MATRIX CCNTAINED IN AN ARRAY OF DIM(IX1,IX2)
C     Y=M-VECTOR CONTAINED IN AN ARRAY OF DIM(IX1)
C     B=N-VECTOR CONTAINED IN AN ARRAY OF DIM(IX2)
C     XX,XXI=WORK ARRAYS OF DIM(IX2,IX2)
C     WY=WORK ARRAY OF DIM(IX1)
C     WX=WORK MATRIX OF DIM(IX1,IX2)
C     XY=WORK ARRAY OF DIM(IX2)
C     WK=WORK ARRAY OF DIM(N**2 + 3*N)    OR LARGER
C
      REAL*4 YS(8),XS(8,4),BS(4)
      REAL*8 Y(8),X(8,4),B(4),XTX(4,4),XTY(4)
C****** CONVERT  REAL*4 TO REAL*8 ***********************************
      DO 10 I=1,IX1
         Y(I)=DBLE(YS(I))
         DO 5 J=1,IX2
```

```
          X(I,J)=DBLE(XS(I,J))
    5   CONTINUE
   10 CONTINUE
C
      CALL MATSQ (X,XTX,M,N)
      CALL MATMUL (X,Y,XTY,M,N)
      CALL CHOLES (XTX,XTY,B,N)
C
C****** CONVERT  REAL*8 TO REAL*4 ****************************************
        DO 15 J=1,IX2
          BS(J)=SNGL(B(J))
   15   CONTINUE
C
      RETURN
      END
C
C************************************************************************
C
      SUBROUTINE NUMPRT(IX,J,IK,PLOT)
C
C     NUMPRT PLOTS THE NUMBER IK IN THE 2-D ARRAY PLOT CENTERED ON
C     THE PLOT(IX,J) POSITION.
C     IX = COLUMN OF MATRIX PLOT WHERE NUMBER IS TO BE PRINTED.
C     J = ROW OF MATRIX WHERE NUMBER IS TO BE PRINTED.
C     IK = NUMBER TO BE PRINTED
C     PLOT = 2-D ARRAY WHERE NUMBER IS TO BE PLOTED.
C
      INTEGER*2 NUM(10),PLOT(122,50)
      DATA NUM/'0','1','2','3','4','5','6','7','8','9'/
      IF (IK.LT.10) GO TO 1
      IF (IK.LT.100) GO TO 2
      IF (IK.LT.1000) GO TO 3
      IF (IK.LT.10000) GO TO 4
      I10000 = IK/10000
      PLOT(IX-2,J) = NUM(I10000+1)
      I1000 = (IK-I10000*10000)/1000
      PLOT(IX-1,J) = NUM(I1000+1)
      I100 = (IK-I10000*10000-I1000*1000)/100
      PLOT(IX,J) = NUM(I100+1)
      I10 = (IK-I10000*10000-I1000*1000-I100*100)/10
      PLOT(IX+1,J) = NUM(I10+1)
      I1 = (IK-I10000*10000-I1000*1000-I100*100-I10*10)
      PLOT(IX+2,J) = NUM(I1+1)
      GO TO 22
    4 I1000 = IK/1000
      PLOT(IX-2,J) = NUM(I1000+1)
      I100 = (IK-I1000*1000)/100
      PLOT(IX-1,J) = NUM(I100+1)
```

```
      I10 =  (IK-I1000*1000-I100*100)/10
      FLOT(IX,J) = NUM(I10+1)
      I1 =  (IK-I1000*1000-I100*100-I10*10)
      FLOT(IX+1,J) = NUM(I1+1)
      GO TO 22
    3 I100 = IK/100
      FLOT(IX-1,J) = NUM(I100+1)
      I10 =  (IK-I100*100)/10
      FLOT(IX,J) = NUM(I10+1)
      I1 =  (IK-I100*100-I10*10)
      FLOT(IX+1,J) = NUM(I1+1)
      GO TO 22
    2 I10 = IK/10
      FLOT(IX-1,J) = NUM(I10+1)
      I1 =  (IK-I10*10)
      FLOT(IX,J) = NUM(I1+1)
      GO TO 22
    1 FLOT(IX,J) = NUM(IK+1)
   22 RETURN
      END
C
C*****************************************************************
C
      SUBROUTINE SECEST(X,N,M,NEK,EST,Y,KF)
      REAL X(50000),Y(12500)
C     COMPUTE ESTIMATES "EST" FOR SECTION LENGTH NEK
      NBK=N/NEK
      KF=0
      DO 10 I=1,M
       IP= (I-1)*N + 1
       DO 15 J=1,NBK
        KP=KF+1
        Y(KP)=EST(X(IP),NEK)
        IP=IP+NEK
   15  CONTINUE
   10 CONTINUE
      RETURN
      END
C
C*****************************************************************
C
      SUBROUTINE MAXMIN(Y,N,YMAX,YMIN)
C     RETURNS MAX AND MIN VALUES OF VECTOR Y OF LENGTH N
      REAL Y(N)
      YMAX=Y(1)
      YMIN=Y(1)
      DO 605 J=1,N
       IF(Y(J).LT. YMIN) YMIN=Y(J)
```

```
      IF(Y(J).GT. YMAX) YMAX=Y(J)
  605 CONTINUE
      RETURN
      END
C
C***********************************************************************
C
      FUNCTION PCTL(Y,N,P)
C     COMPUTES P PERCENTILE OF N VALUES IN Y
      REAL Y(N)
      R=P*FLOAT(N+1)
      CALL SORT(Y,N)
      I=MAX0(INT(R) , 1)
      I=MIN0(I,N)
      J=MIN0(INT(R+1.)  , N)
      R=R-INT(R)
      PCTL=Y(I)+R*(Y(J)-Y(I))
      RETURN
      END
C
C***********************************************************************
C
      SUBROUTINE DELETO(Y,KP,YMAX,YMIN)
C     SUBROUTINE SCALES THE GRAPH TO UPPER (LOWER) QUARTILE + (-)
C     1.5 TIMES INTERQUARTILE DISTANCE OR TO FIRST POINT WITHIN
C     THESE LIMITS IF NO POINTS EXIST OUTSIDE.
      REAL Y(KP),Z(12500)
C     COPY BEFORE SORTING
      DO 23 I=1,KP
         Z(I) = Y(I)
   23 CONTINUE
      P25 =PCTL(Z,KP,.25)
      P75 =PCTL(Z,KP,.75)
      P50 =PCTL(Z,KP,.50)
      YMIN=2.5*P25-1.5*P75
      YMAX=2.5*P75-1.5*P25
      IF(Z(1).GT.YMIN)  YMIN=Z(1)
      IF(Z(KP).LT.YMAX)  YMAX=Z(KP)
      RETURN
      END
C
C***********************************************************************
C
      SUBROUTINE CHOLES (XTX,XTY,BHAT,N)
      REAL*8 L(4,4),SUM,LT(4,4),XTX(4,4),XTY(4),BHAT(4),WY(4)
      REAL*4 B(4)
      INTEGER P
C
```

```
C****** INIT L ************************************
      DC 100 I=1,N
       BHAT(I)=0.CDO
       DO 50 J=1,N
        L(I,J)=0.0DO
        LT(I,J)=C.0DO
  50   CONTINUE
 100  CCNTINUE
C*****  ALGORITHM  DECOMPOSITION  ****************
      L(1,1)=DSQRT(XTX(1,1))
      DC 500 K=2,N
       KK=K-1
       DO 200 J=1,KK
        JJ=J-1
        SUM=0.0D0
        IF (J.EQ.1) GO TO 150
         DO 140 P=1,JJ
          SUM=SUM+(L(K,P)*L(J,P))
 140     CONTINUE
 150    CONTINUE
        L(K,J)=(XTX(K,J)-SUM)/L(J,J)
 200   CCNTINUE
       SUM=0.0D0
       DC 300 P=1,KK
        SUM=SUM+(L(K,P)**2)
 300   CCNTINUE
       L(K,K)= DSCRT (XTX(K,K)-SUM)
 500  CCNTINUE
C  BUILD L-TRANSPCSE  IN  IT ********************
      DC 540 I=1,N
       DO 530 J=1,N
        LT(I,J)=L(J,I)
  530   CONTINUE
  540 CCNTINUE
C
C******  A L G O R I T H M   PART 1 A. 2 *********
C******  L * WY = XTY
      WY(1)=XTY(1)/L(1,1)
      DO 700 I=2,N
         II=I-1
         SUM=0.0DC
         DO 600 J=1,II
            SUM=SUM+(WY(J)*L(I,J))
 600     CONTINUE
         WY(I)=(XTY(I)-SUM)/L(I,I)
 700  CCNTINUE
C
C ***  IT * BHAT = WY  ***************************
```

```
      BHAT(N)=WY(N)/LT(N,N)
      DO 800 II=2,N
         I=N-II+1
         SUM=0.0D0
         DO 750 J=I,N
            SUM=SUM+(BHAT(J)*LT(I,J))
 750     CONTINUE
         BHAT(I) = (WY(I) -SUM)/LT(I,I)
 800  CONTINUE
C
      DO 950 I=1,4
         B(I)=SNGL(BHAT(I))
 950  CONTINUE
C
      RETURN
      END
C
C******    MATRIX MULTIPLICATION   XT * X = XRES   **
C
      SUBROUTINE MATSQ ( X, XRES, M, N )
      REAL*8 X(8,4),XT(4,8),XRES(4,4),SUM
C
C ***  BUILD  X-TRANSPOSE IN LT  *****************
      DO 20 I=1,M
         DO 10 J=1,N
            XT(J,I)=X(I,J)
 10      CONTINUE
 20   CONTINUE
C
C ***   XT * X = XRES    ****
C
      DO 50 I=1,N
         DO 40 J=1,N
            SUM=0.0D0
            DO 30 K=1,M
               SUM=SUM+(XT(I,K)*X(K,J))
 30            CONTINUE
               XRES (I,J) = SUM
 40      CONTINUE
 50   CONTINUE
      RETURN
      END
C
C  ****  MATRIX MULTIPLICATION  XT * Y = XTY   ****
C
      SUBROUTINE MATMUL ( X,Y,XTY,M,N )
      REAL*8 Y(8),XT(4,8),X(8,4),XTY(4),SUM
C
```

```
C*****  BUILD XT  ********************************
      DO 20 I=1,M
         DO 10 J=1,N
            XT(J,I)=X(I,J)
 10      CONTINUE
 20   CONTINUE
C
C*****  XT * Y = XTY  ****************************
C
      DO 50 I=1,M
         SUM=0.0D0
         DO 40 J=1,N
            SUM=SUM+(XT(I,J)*Y(J))
 40      CONTINUE
         XTY(I)=SUM
 50   CONTINUE
      RETURN
      END
C
C*****************************************************************
C
      SUBROUTINE SORT (Y,N)
C   INPLACE SORT USING SHELL ALGORITHM ***********
      REAL Y(N),TEMP
      INTEGER GAP
      LOGICAL EXCH
C
      GAP=(N/2)
  5   IF (.NOT.(GAP.NE.0)) GO TO 500
 10      CONTINUE
            EXCH=.TRUE.
            K=N-GAP
            DO 200 I=1,K
            KK=I+GAP
            IF(.NOT.(Y(I).GT.Y(KK))) GO TO 100
               TEMP=Y(I)
               Y(I)=Y(KK)
                  Y(KK)=TEMP
                  EXCH=.FALSE.
 100        CONTINUE
 200        CONTINUE
         IF (.NOT.(EXCH)) GO TO 10
         GAP=(GAP/2)
      GO TO 5
 500  CONTINUE
C
      RETURN
      END
```

APPENDIX B

SIMTB2 (VERSION 2) PROGRAM LISTING

```
C
C                     SIMTB2  PROGRAM  LISTING
C
C
C
C  PURPOSE     TO GENERATE REGRESSION ADJUSTED ESTIMATES AND BOX PLOTS
C              OF ESTIMATES OF AN INPUT RAW DATA SERIES X CONTAING M
C              (REPLICATIONS) OF N VALUES EACH.  UP TO 3 ESTIMATING
C              FUNCTIONS CAN BE USED.  THE GRAPHS CAN ALL BE OF THE SAME
C              SCALE OR SCALED INDIVIDUALLY.
C
C
C  DESCRIPTION OF PARAMETERS
C
C     X        REAL*4 ARRAY CONTAINING DATA.
C              A MAXIMUM OF 50,000 DATA ELEMENTS CAN BE STORED IN X.
C
C     N        NUMBER OF DATA ELEMENTS PER SECTION (N IS SAMPLE SIZE).
C              N CANNOT EXCEED 50,000 AND M*N MUST NOT EXCEED 50,000.
C
C     M        NUMBER OF SECTIONS (REPLICATIONS).
C              M CANNOT EXCEED 100 AND M*N MUST NOT EXCEED 50,000.
C
C     NE       INTEGER ARRAY OF SIZE 8 CONTAINING SUBSAMPLE SIZES FOR N.
C              THE VALUES OF NE MUST BE FROM SMALLEST TO LARGEST.
C              NO ELEMENT OF THE ARRAY NE CAN BE GREATER THAN N.
C              M*(N/NE(1)) MUST NOT EXCEED 12,500.
C
C     L        NUMBER OF SUBSAMPLE SIZES FROM NE(8) THAT WILL BE USED TO
C              SECTION N.
C              IT IS ALSO THE NUMBER OF BOXPLOTS THAT WILL BE PRODUCED.
C
C     D        DEGREE OF REGRESSION FOR MEAN AND VARIANCE REGRESSIONS.
C              D WILL BE REDUCED BY RAGE IF THE SAMPLE IS NOT LARGE
C              ENOUGH. D MUST BE 1,2 OR 3. D=0 WILL IGNORE REGRESSIONS.
C
C              *** SCALING ***
C                 SCALING IS ACCOMPLISHED BY TAKING THE SMALLEST AND THE
C                 LARGEST ESTIMATE VALUES FROM ALL ESTIMATING FUNCTIONS
C                 AND FROM FOR ALL SUBSAMPLE SIZES.
C                    THE SEI PARAMETER ALLOWS THE USER TO SCALE THE GRAPH
C                 OF EACH ESTIMATOR INDIVIDUALLY OR TO SCALE THEM ALL TO
C                 THE SAME SCALE.  SCALING ALL TO THE SAME SCALE IS
C                 ACCOMPLISHED BY TAKING THE MINIMUM AND MAXIMUM  ESTIMAT
C                 FROM ALL THE ESTIMATORS USING NE(1) SUBSAMPLE SIZE.
C                    THE RG PARAMETER ALLOWS THE USER TO RREDUCE THE VER-
C                 TICAL SCALE TO; THE UPPER QUARTILE DISTANCE + 1.5 TIMES
C                 INTERQUARTILE DISTANCE AS THE MAX VALUE AND THE LOWER
```

```
C                 QUARTILE - 1.5 TIMES THE INTERQUARTILE DISTANCE AS THE
C                 MIN VALUE.  THE INTERQUARTILE DISTANCE IS COMPUTED FROM
C                 THE SAMPLE OF ESTIMATES FROM THE NE(1) SUBSAMPLE SIZE.
C                 IF THERE ARE NO ESTIMATES OUTSIDE THESE MIN AND MAX
C                 VALUES THEN THE SCALE IS TO THE FIRST VALUE WITHIN.
C                 IF THERE ARE ESTIMATES OUTSIDE THESE LIMITS THEN THEY
C                 ARE COUNTED AND THE NUMBER PRINTED AT THE ENDS OF THE
C                 BOX PLOTS.
C                    THE SVS PARAMET ALLOWS THE USER TO SET THE VERTICAL
C                 SCALE. WHEN THE VERTICAL SCALE IS SET THE SEI PARAMETER
C                 IS IGNORED AND THE VERTICAL SCALE BECOMES YMIN AND YMAX
C
C     RG        RG=0 DO NOT REDUCE THE VERTICAL SCALE OF THE GRAPHS.
C               RG=1 REDUCE GRAPHICS VERTICAL SCALE TO UPPER (LOWER)
C                    QUARTILE + (-) INTERQUARTILE DISTANCE.
C
C     SEI       SEI=0 DO NOT SCALE ESTIMATORS' GRAPHS INDIVIDUALLY.
C               SEI=1 SCALE ESTIMATORS' GRAPHS INDIVIDUALLY.
C
C     SVS       SVS=0  PROGRAM WILL CALCULATE VERTICAL SCALE.
C               SVS=1  USER SETS VERTICAL SCALE TO YMIN AND YMAX.
C
C     YMIN      LOW VALUE OF VERTICAL SCALE. SET BY USER WHEN SVS=1
C
C     YMAX      HIGH VALUE OF VERTICAL SCALE. SET BY USER WHEN SVS=1
C
C     NEST      NUMBER OF ESTIMATORS THAT WILL BE USED TO CALCULATE
C               STATISTICAL PARAMETER FROM X DATA.
C               NEST MUST BE 1,2 OR 3.
C
C     EST1      NAMES OF THE ESTIMATOR FUNCTIONS THAT WILL BE USED TO
C     EST2      CALCULATE THE STATISTICAL PARAMETER.
C     EST3      CALL SEQUENCE ON EACH FUNCTION IS: CALL FNAME(X,N) WHERE
C               X IS THE DATA ARRAY AND N IS THE NUMBER OF DATA POINTS.
C               THEY MUST BE DECLARED IN THE CALLING PROGRAM (RAGE) IN
C               THE ORDER THEY ARE USED. DUMMY VARIABLES MUST BE INSERTED
C               WHEN THERE ARE LESS THAN 3 ESTIMATORS.
C
C     TTL1      TITLES ASSOCIATED WITH EACH ESTIMATOR (EST1,2,3).  A MAX
C     TTL2      OF 120 CHARACTERS CAN BE USED TO DESCRIBE EACH ESTIMATOR.
C     TTL3      EACH TITLE MUST BE DECLARED AS REAL*8(15) ARRAYS UNLESS
C               PASSED AS AN ARGUMENT OF THE CALLING PROGRAM RAGE.
C               WHEN PASSING THE TITLE AS AN ARGUMENT THERE  MUST BE A
C               MINIMUM OF 120 CHARACTERS BETWEEN APOSTROPHES.
C
C*********************************************************************
C
```

```
      SUBROUTINE SIMTB(X,N,M,NE,L,D,RG,SEI,SVS,YMIN,YMAX,NEST,EST1,
     *TTL1,EST2,TTL2,EST3,TTL3,IR,IRK)
      REAL X(IR,IRK),ULH(4),Y(12500)
      REAL*8 TTL1(15),TTL2(15),TTL3(15)
      INTEGER NE(8),RG,SEI,SVS,SM
      INTEGER D,I,NEST,TEST
C
      SM=NE(1)
      MN = M*N
      LT=L-1
      IF(LT.EQ.0) GO TO 13
      DO 11 I=1,LT
         I1=I+1
         IF(NE(I).GT.NE(I1)) WRITE(6,110)
   11 CONTINUE
   13 TEST=0
      IF(NEST.EQ.1.OR.NEST.EQ.2.OR.NEST.EQ.3) GO TO 1
      WRITE(6,106)
      TEST=1
    1 IF(MN.LE.50000) GO TO 2
      WRITE(6,105)
      TEST=1
    2 IF(M.GE.1.AND.M.LE.100) GO TO 3
      WRITE(6,104)
      TEST=1
    3 IF(L.GE.1.AND.L.LE.8) GO TO 4
      WRITE(6,103)
      TEST=1
    4 IF(D.LE.3) GO TO 5
      WRITE(6,108)
      TEST=1
    5 K=N/NE(L)
      IF(K.GE.1) GO TO 6
      WRITE(6,107)
      TEST=1
    6 K=M*(N/NE(1))
      IF(K.LE.12500) GO TO 7
      WRITE(6,109)
      TEST=1
    7 IF (TEST.NE.0) GO TO 80
      ULH(2)=YMIN
      ULH(4)=YMAX
C     DETERMINE HOW EACH GRAPH IS TO BE SCALED.
      IF (SVS .EQ. 1) GO TO 50
      IF (SEI .EQ. 1) GO TO 75
C
C     ****************************************************************
C     *GRAPH ALL ESTIMATORS TO THE SAME SCALE OF ESTIMATOR W/WIDEST PTS
```

```
C     ***************************************************************
      ULH(2)=1.E30
      ULH(4)=-1.E30
      DO 10 IK=1,I
C       FIND VERTICAL SCALE FOR 1ST ESTIMATOR.
        CALL SECEST(X,N,M,NE(IK),EST1,Y,KP,IR,IRK)
        IF(RG.EQ.1) CALL DELETO(Y,KP,YMAX,YMIN)
        IF(RG.NE.1) CALL MAXMIN(Y,KP,YMAX,YMIN)
        IF(YMIN .LT. ULH(2)) ULH(2)=YMIN
        IF(YMAX .GT. ULH(4)) ULH(4)=YMAX
C
C       FIND VERTICAL SCALE FOR 2ND ESTIMATOR.  KEEP WIDEST PAIR****
        IF(NEST .LT. 2) GO TO 10
        CALL SECEST(X,N,M,NE(IK),EST2 ,Y,KP,IR,IRK)
        IF(RG.EQ.1) CALL DELETO(Y,KP,YMAX,YMIN)
        IF(RG.NE.1) CALL MAXMIN(Y,KP,YMAX,YMIN)
        IF(YMIN .LT. ULH(2)) ULH(2)=YMIN
        IF(YMAX .GT. ULH(4)) ULH(4)=YMAX
C
C       FIND VERTICAL SCALE FOR 3RD ESTIMATOR.  KEEP WIDEST PAIR****
        IF(NEST .LT. 3) GO TO 10
        CALL SECEST(X,N,M,NE(IK),EST3 ,Y,KP,IR,IRK)
        IF(RG.EQ.1) CALL DELETO(Y,KP,YMAX,YMIN)
        IF(RG.NE.1) CALL MAXMIN(Y,KP,YMAX,YMIN)
        IF(YMIN .LT. ULH(2)) ULH(2)=YMIN
        IF(YMAX .GT. ULH(4)) ULH(4)=YMAX
   10 CONTINUE
C     RETURN CALCULATED SCALE TO CALLER
      YMIN=ULH(2)
      YMAX=ULH(4)
C
C     PROCESS BOXPLOTS USING FIXED VERTICAL SCALE IN VECTOR ULH
C     ONE CALL FOR EACH ESTIMATOR USED.
   50 CALL PRST(X,N,M,EST1,NE,L,RG,D,ULH,Y,IR,IRK)
      WRITE(6,101) ULH(2),ULH(4)
      WRITE(6,102) TTL1
      IF (NEST. LT. 2) GO TO 80
      CALL PRST(X,N,M,EST2,NE,L,RG,D,ULH,Y,IR,IRK)
      WRITE(6,101) ULH(2),ULH(4)
      WRITE(6,102) TTL2
      IF (NEST. LT. 3) GO TO 80
      CALL PRST(X,N,M,EST3,NE,L,RG,D,ULH,Y,IR,IRK)
      WRITE(6,101) ULH(2),ULH(4)
      WRITE(6,102) TTL3
      GO TO 80
C     ***************************************************
C     *GRAPH EACH ESTIMATOR SCALED TO ITS WIDEST POINTS.*
C     ***************************************************
```

```
C     FIND VERTICAL SCALE FOR 1ST ESTIMATOR AND GRAPH.
   75 CALL SECEST(X,N,M,NE(1),EST1,Y,KP,IR,IRK)
      IF(RG.EQ.1) CALL DELETO(Y,KP,YMAX,YMIN)
      IF(RG.NE.1) CALL MAXMIN(Y,KP,YMAX,YMIN)
      ULH(2)=YMIN
      ULH(4)=YMAX
      CALL PRST(X,N,M,EST1,NE,L,RG,D,ULH,Y,IR,IRK)
      WRITE(6,101) ULH(2),ULH(4)
      WRITE(6,102) TTL1
      IF(NEST .LT. 2) GO TO 80
C
C     FIND VERTICAL SCALE FOR 2ND ESTIMATOR AND GRAPH.
      CALL SECEST(X,N,M,NE(1),EST2 ,Y,KP,IR,IRK)
      IF(RG.EQ.1) CALL DELETO(Y,KP,YMAX,YMIN)
      IF(RG.NE.1) CALL MAXMIN(Y,KP,YMAX,YMIN)
      ULH(2)=YMIN
      ULH(4)=YMAX
      CALL PRST(X,N,M,EST2,NE,L,RG,D,ULH,Y,IR,IRK)
      WRITE(6,101) ULH(2),ULH(4)
      WRITE(6,102) TTL2
      IF(NEST .LT. 3) GO TO 80
C
C     FIND VERTICAL SCALE FOR 3RD ESTIMATOR AND GRAPH.
      IF (SVS.EQ.1) GO TO 78
      CALL SECEST(X,N,M,NE(1),EST3 ,Y,KP,IR,IRK)
      IF(RG.EQ.1) CALL DELETO(Y,KP,YMAX,YMIN)
      IF(RG.NE.1) CALL MAXMIN(Y,KP,YMAX,YMIN)
      ULH(2)=YMIN
      ULH(4)=YMAX
   78 CALL PRST(X,N,M,EST3,NE,L,RG,D,ULH,Y,IR,IRK)
      WRITE(6,101) ULH(2),ULH(4)
      WRITE(6,102) TTL3
C
   80 CONTINUE
  102 FORMAT(1X,'ESTIMATOR: ',15A8)
  101 FORMAT(1X,'VERTICAL SCALE:  YMIN =',F10.4,/,18X,'YMAX =',F10.4/)
  103 FORMAT(' *** ERROR...L MUST BE AN INTEGER BETWEEN 1 AND 8. ***')
  104 FORMAT(' *** ERROR...M MUST BE AN INTEGER BETWEEN 1 AND 100. ***'
  105 FORMAT(' *** ERROR...M*N EXCEEDS 50,000. ***')
  106 FORMAT(' *** ERROR...NEST MUST BE 3 OR LESS. ***')
  107 FORMAT(' *** ERROR...N/NE(L) MUST BE 1 OR GREATER TO COMPUTE',
     +' STATISTICS.',/,'     INCREASE N OR DECREASE NE(L). ***')
  108 FORMAT(' *** ERROR...D MUST BE LESS THAN OR EQUAL TO 3. ***')
  109 FORMAT(' *** ERROR...M*(N/NE(1)) MUST NOT EXCEED 12,500. ***')
  110 FORMAT(' *** WARNING... NE ARRAY ELLEMENTS ARE NOT IN ORDER OF '/
     +'      INCREASING SIZE. IF NE(1) IS NOT SMALLEST ELEMENT, SCALING'
     +'      MAY CAUSE POINTS TO FALL OUTSIDE RANGE OF SCALE.')
      RETURN
```

```
      END
C*********************************************************************
C
      SUBROUTINE ERST(X,N,M,EST,NE,L,RG,UD,ULH,Y,IR,IRK)
C     REGRESSION ADJUSTED ESTIMATE
C     CALCULATES ESTIMATES FROM USER DATA USING "EST" FUNCTION
C     PLOTS BASIC OR RETRENCHED GRAPH ON LINE PRINTER
C
C     EST = NAME OF USER WRITTEN ESTIMATING FUNCTION.
C           USAGE:     FUNCTION_NAME(X,N)
C           WHERE X IS A VECTOR WITH N ENTRIES
C
C     M = NO. OF REPLICATIONS (MUST BE <= 100)
C     N = NUMBER OF VALUES IN EACH REPLICATION (M*N MUST BE <= 10000)
C     X = USERS VECTOR WITH M CONSECUTIVE BATCHES OF N VALUES EACH
C     L = NO. OF SECTION SIZES (MUST BE BETWEEN 1 AND 8)
C     NE = ARRAY WITH THE L SUBSAMPLE SIZES (MUST BE IN ASCENDING ORDER
C     UD = DEGREE OF THE REGRESSION (MUST BE <= 3 & <= L-1 )
C     ULH= XMIN,YMIN,XMAX,YMAX  IN USER UNITS
C          ONLY ULH(2) AND ULH(4) NEED TO BE PASSED. OTHERS CALC. HERE
C
      DIMENSION  NE(L)
      INTEGER NB(8),LOCX(8),D1,IWIDTH,UD,D,LT,CT,RG,RNEK
      INTEGER*2 PLOT(122,50), CBAR,BLK,DASH,CSTR,NUM(10),DOT
      REAL*4 X(IR,IRK),ULH(4),DLH(4),Y(12500)
      REAL*4 RH(8,100),STAT(8,6),VT(8)
      REAL*8 SUM2,SUM3,SUM4,LABEL(5)
      REAL*4 RA(8,4),RV(8,4),B(4,100),V(4),BA(4),BV(4),BS(4),RT(8),LT(8
C
      DATA DLH/1.,1.,122.,50./
      DATA BLK/' '/, DASH/'-'/, CBAR/'|'/, DOT/'.'/
      DATA LABEL/'MEAN','STD','STD MEAN','SKEWNESS','KURTOSIS'/
      D=MIN0(3,UD,L-1)
      IX1=8
      IX2=4
      MN=M*N
      D1=D+1
      IWIDTH= IFIX(DLH(3))
C     BUILD REGRESSION MATRICES FOR AVERAGES AND VARIANCES
      DO 84 K=1,L
       DO 86 J=1,D1
        T=FLOAT(NE(L))/FLOAT(NE(K))
        RA(K,J)=T**(J-1)
        RV(K,J)=T**((FLOAT(J))/2.0)
   86  CONTINUE
   84 CONTINUE
C     CLEAR PLOT ARRAY
      DO 3 J=1,50
```

```
      DO 4  I=1,122
       PLOT(I,J)=BLK
    4  CONTINUE
    3 CONTINUE
C     SET HORIZONTAL XMIN, XMAX
      ULH(1)=.7*NE(1)
      ULH(3)=1.2*NE(L)
C     SET SCALE
      CALL SCALE(ULH,DLH)
C     COMPUTE LOCATION OF BOXPLOTS ALONG X-AXIS
      LAST=-1
      DO 5  K=1,L
       NE(K)=N/NE(K)
       IOCX(K)=(NE(K)-ULH(1))*(DLH(3)-DLH(1))/(ULH(3)-ULH(1))+DLH(1)+.5
       IF(LOCX(K).LT. LAST+4)  LOCX(K)=LAST+4
       LAST=LOCX(K)
    5 CONTINUE
C
      DO 80  K=1,L
       NEK=NE(K)
       RNEK=NE(K)
C      SECTION 6 COMPUTE ESTIMATORS FOR SIZE K
       CALL SECEST(X,N,M,RNEK,EST,Y,KP,IR,IRK)
C      AVERAGE ESTIMATES OF SIZE NE(K) FOR EACH OF M REPLICATIONS
       KP=0
       DO 10 I=1,M
        RH(K,I)=0.
        DO 15 J=1,NBK
         KP=KP+1
         RH(K,I)=RH(K,I)+Y(KP)
   15   CONTINUE
        RH(K,I)=RH(K,I)/FLOAT(NBK)
   10  CONTINUE
      CALL BOXPRT(Y,KP,IOCX(K),PLOT,RG)
      IF (K .GT. 8) GO TO 80
C     COMPUTE MEAN AND MOMENT ESTIMATES
      XMEAN=0.
      DO 180 IM1=1,KP
       XMEAN=XMEAN+Y(IM1)
  180 CONTINUE
      XMEAN=XMEAN/FLOAT(KP)
      SUM2 = 0.0D0
      SUM3 = 0.0D0
      SUM4 = 0.0D0
      DO 190 IP1=1,KP
      DEV = Y(IP1) - XMEAN
      SUM2 = SUM2 + DEV * DEV
      SUM3 = SUM3 + DEV ** 3
```

```
      SUM4 = SUM4 + DEV ** 4
  190 CCNTINUE
C
C     CHECK TO INSURE SAMPLE SIZE IS LARGE ENOUGH FCR
C     EACH MOMENT COMPUTATION.
      IF (KP.LT.2) GO TO 7
      VAR = SUM2 / (KP - 1.0)
      STDV = SQRT(VAR)
    7 IF (KP.LT.3) GO TO 8
      XSUM3 = SNGL(SUM3) * KP /( (KP-1.) * (KP-2.))
      SKEW = XSUM3 / STDV ** 3
    8 IF (KP.LT.4) GO TO 9
      XSUM4 = SNGL(SUM4)*((KP-2.) *KP+3.)/((KP-1.)*(KP-2.)*(KP-3.))
      XSUM4 = XSUM4 - VAR*VAR*3.*(KP-1.)*(KP+KP-3.)/(KP*(KP-2.)*(KP-3))
      CKURT = XSUM4 / (VAR * VAR) - 3.
    9 STAT(K,1)=XMEAN
      STAT(K,2)=STDV
      STAT(K,3)=STDV/SQRT(FLOAT(KP))
      STAT(K,4)=SKEW
      STAT(K,5)=CKURT
      STAT(K,6)=VAR
   80 CCNTINUE
C
C     IF D1.LT.2 THEN NO REGRESSIONS OR PLOTING CAN BE DONE
C
      IF(D1.LT.2) GO TO 113
      DC 92 K=1,M
         DO 47 J=1,L
            BT(J)=BH(J,K)
  47  CCNTINUE
      CALL RREG(HA,RT,BT,L,D1,IX1,IX2)
      B(1,K)=BT(1)
      DO 23 KT=2,D1
         B(KT,K)=BT(KT)*NE(L)**(KT-1)
   23 CCNTINUE
   92 CCNTINUE
C
C     AVERAGE REGRESSION CCEFF. OVER M REPLICATICNS & CALC. VARIANCE
C
      DO 94 I=1,D1
       BA(I)=0.
       BV(I)=0.
       DO 95 J=1,M
        BA(I)=BA(I)+B(I,J)
        BV(I)=BV(I)+B(I,J)**2
   95  CONTINUE
       BA(I)=BA(I)/FLOAT(M)
       IF(M.EQ.1)GO TO 94
```

```
      EV(I)=(BV(I)-M*BA(I)**2)/(M*(M-1.))
      BS(I)=BV(I)**.5
   94 CCNTINUE
C
C     ESTABLISH REGRESSION LINE & ASYMPTOTE
C
      DC 98 I=3,IWIDTH
C      MAP I FROM DEVICE SPACE TO USER SPACE
       UX=(I-DLH(1))*(ULH(3)-ULH(1))/(DLH(3)-DLH(1)) + ULH(1)
C      COMPUTE THE Y VALUE FROM X AND THE REGRESSION COEFFICIENTS.
       UY=BA(1)
       DO 99 J=1,D
        UY=UY+BA(J+1)/UX**J
   99  CONTINUE
C
C      MAP THE Y VALUE FROM USER SPACE TO DEVICE SPACE
       J=(UY-ULH(2))*(DLH(4)-DLH(2))/(ULH(4)-ULH(2)) + DLH(2) + .5
       IF(J .LT. 1 .OR. J .GT. 50) GO TO 98
       IF(PLOT(I,J) .EQ. BLK) PLOT(I,J)=DOT
   98 CCNTINUE
C
C     SCALE ASYMPTOTE, BETA0, AND PLOT ACROSS PLOT.
      J=(BA(1)-ULH(2))*(DLH(4)-DLH(2))/(ULH(4)-ULH(2)) + DLH(2) + .5
      IF(J .LT. 1 .OR. J .GT. 50) GO TO 117
      DC 120 I=3,IWIDTH
       IF(PLCT(I,J) .EQ. BLK) PLOT(I,J)=DASH
  120 CCNTINUE
C
C     REGRESSION CN VARIANCES FROM EACH SEGMENT WITH A VARIANCE.
C
  117 K=M*(N/NE(L))
      LT=L
      DT=D1
      DC 111 I=1,L
         IF(K.GE.2) GO TO 112
         LT=LT-1
         K=M*(N/NE(LT))
  111 CCNTINUE
  112 IF(LT.LT.DT) DT=LT
      IF(DT.LT.2) GO TO 113
      DC 48 J=1,L
         VT(J)=STAT(J,6)*(NE(J)**0.5)
   48 CCNTINUE
      CALL RREG(EV,VT,V,LT,DT,IX1,IX2)
      DC 77 I=1,LT
         V(I) = V(I)*NE(L)**(FLOAT(I)/2.)
   77 CCNTINUE
C
```

```
C       PLOT    *************
C
  113 WRITE(6,102)
      WRITE(6,161) N,M,D
      WRITE(6,101)
      DO 90 J=1,50
       K=51-J
       IF(MOD(K,5).NE.0) GO TO 85
       YLABEL=(K-DLH(2))*(ULH(4)-ULH(2))/(DLH(4)-DLH(2)) + ULH(2)
       WRITE(6,103)  YLABEL,(PLOT(I,K),I=1,IWIDTH)
       GO TO 90
   85  CONTINUE
       WRITE(6,100)   (PLOT(I,K),I=1,IWIDTH)
   90 CONTINUE
C     LABEL X-AXIS BY REUSING PLOT MATRIX.
      DO 115 I=1,122
        PLOT(I,1)=DASH
        PLOT(I,2)=BLK
  115 CONTINUE
      DO 130 J=1,L
       PLOT(LOCX(J),1)=CBAR
       IK =  NE(J)
       IX =  LOCX(J)
       CALL  NUMPRT(IX,2,IK,PLOT)
  130 CONTINUE
      WRITE(6,106)  (PLOT(I,1),I=1,IWIDTH)
      WRITE(6,104)  (PLOT(I,2),I=1,IWIDTH)
      L8=L
      IF(L8 .GT. 8) L8=8
      WRITE(6,156)
      WRITE(6,146)  (NE(I),I=1,L8)
      WRITE(6,157)  LABEL(1),  (STAT(K,1),K=1,L8)
C     CHECK TO INSURE SAMPLE SIZE IS LARGE ENOUGH TC COMPUTE
C     STATISTICS BEFORE ATTEMPTING TO PRINT STATS.
      LT=L8
      L1=L8
      DO 21  I=1,IT
         K1  = M*(N/NE(L1))
         IF  (K1.GE.2) GO TO 11
         L1  = L1-1
   21 CONTINUE
      GO TO  14
   11 WRITE(6,157) LABEL(2),  (STAT(K,2),K=1,L1)
      WRITE(6,157) LABEL(3),  (STAT(K,3),K=1,L1)
      LT =  L1
      DO 22  I=1,IT
         K1  = M*(N/NE(L1))
         IF  (K1.GE.3) GO TO 12
```

```
         L1 = L1-1
   22 CONTINUE
      GO TO 14
   12 WRITE(6,158) LABEL(4),  (STAT(K,4),K=1,L1)
      LT = L1
      DO 33  I=1,IT
         K1 = M*(N/NE(L1))
         IF  (K1.GE.4) GO TO 13
         L1 = L1-1
   33 CONTINUE
      GO TO 14
   13 WRITE(6,158) LABEL(5),  (STAT(K,5),K=1,L1)
      IF(D1.LT.2) GO TO 444
      WRITE(6,151)  (BA(I),I=1,D1)
   14 IF(M.LT.2)GO TO 444
      WRITE(6,152)  (BV(I),I=1,D1)
      WRITE(6,153)  (BS(I),I=1,D1)
  444 IF(DT.LE.1) GO TO 999
      WRITE(6,155)  (V(I),I=1,DT)
  999 WRITE(6,162)
  100 FORMAT(9X,'|',122A1,'|')
  101 FORMAT(9X,'+',122('-'),'+')
  102 FORMAT('1')
  103 FORMAT(' ',G8.2,'-',122A1,'|')
  104 FORMAT(10X,122A1)
  106 FORMAT(9X,'+',122A1,'+')
  151 FORMAT(/,' MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:',4G20.6
  152 FORMAT(' VARIANCE OF REGRESSION - COEFFICIENTS:',8X,4G20.6)
  153 FORMAT(' STD DEV OF REGRESSION - COEFFICIENTS:',9X,4G20.6)
  156 FORMAT(/' SUBSAMPLE')
  146 FORMAT(' SIZE       ',8(I8,6X),/)
  157 FORMAT(1X,A6,8G14.4)
  158 FORMAT(1X,A6,8F14.3)
  159 FORMAT(/,' REGRESSION ON VARIANCE - COEFFICIENTS: ',7X,4G20.6)
  161 FORMAT(9X,'SAMPLE SIZE (N):',I5,20X,'NO. OF REPLICATIONS (M):',
     +I5,18X,'DEGREE OF REGRESSION (D):',I3)
  162 FORMAT(/)
      RETURN
      END
C
C**********************************************************************
C
      SUBROUTINE BOXPRT(Y,NY,IX,PLOT,RG)
C     PREPARES BOXPLOT  FROM VECTOR Y (IN 2-D ARRAY PLOT)
      REAL Y(NY),ULH(4),DLH(4)
      INTEGER RG
      INTEGER*2 PLOT(122,50),DASH,CBAR,CROSS,CSTR,CC,NUM(10)
      LOGICAL*1 LFLAG
```

```
      DATA DASH/'-'/,CBAR/'|'/,CSTR/'*'/,CROSS/'+'/,CC/'0'/
      IF(NY .GE. 9) GO TO 5
C     WHEN LESS THAN 9 POINTS JUST SHOW THE POINTS
      DO 8 I=1,NY
       J=(Y(I)-YMIN)*VSCALE + 1.
C      IGNORE VALUE IF IT FALLS OUTSIDE WINDOW
       IF(J.GT.50  .OR.  J.LT.1) GO TO 8
       PLOT(IX,J)=CO
    8 CONTINUE
      SUM=0.
      DO 88 I=1,NY
       SUM=SUM+Y(I)
   88 CONTINUE
      SUM=SUM/FLOAT(NY)
      MEAN=(SUM-YMIN)*VSCALE+1
      PLOT(IX,MEAN)=CSTR
      GO TO 99
    5 CONTINUE
      LFLAG=.FALSE.
      P25 =PCTL(Y,NY,.25)
      P75 =PCTL(Y,NY,.75)
      P50 =PCTL(Y,NY,.50)
      IQ1=(P25-YMIN)*VSCALE+1.
      IQ2=(P50-YMIN)*VSCALE+1.
      IQ3=(P75-YMIN)*VSCALE+1.
      XLOW=2*P25-P75
      ILOW=(XLOW-YMIN)*VSCALE+1.
      XHI=2*P75-P25
      IHI=(XHI-YMIN)*VSCALE+1.
      CLOW=2.5*P25-1.5*P75
      CHI=2.5*P75-1.5*P25
C     DRAW BOX
      DO 20 I=IQ1,IQ3
       PLOT(IX-1,I)=CBAR
       PLOT(IX+1,I)=CBAR
   20 CONTINUE
      PLOT(IX-1,IQ1)=DASH
      PLOT(IX+1,IQ1)=DASH
      PLOT(IX-1,IQ3)=DASH
      PLOT(IX+1,IQ3)=DASH
C     DETERMINE IF OUTLIERS ARE TO BE COUNTED AND THE NUMBER PRINTED.
      IF (RG.EQ.1) GO TO 55
      DO 30 I=1,NY
       J=(Y(I)-YMIN)*VSCALE+1.
C      IGNORE VALUE IF IT FALLS OUTSIDE WINDOW
       IF(J.GT.50  .OR.  J.LT.1) GO TO 30
       IF(Y(I).LT.CLOW) PLOT(IX,J)=CSTR
       IF(Y(I).LT.CLOW) GO TO 30
```

```
      IF(Y(I).GE.CLOW .AND.Y(I).LT.XLOW) PLOT(IX,J)=CO
      IF(LFLAG .OR. Y(I).LT.XLOW) GO TO 25
C     THIS IS THE LOW-CROSS POINTER (1ST POINT GE XLOW)
      LFLAG=.TRUE.
      ILX=J
C     NEXT LINE ENDS UP WITH HI-CROSS POINTER (LAST POINT LE XHI)
   25 IF(Y(I).LE.XHI) IHX=J
      IF(Y(I).GT.XHI .AND.Y(I).LE.CHI) PLOT(IX,J)=CO
      IF(Y(I).GT.CHI) PLOT(IX,J)=CSTR
   30 CONTINUE
      GO TO 56
C
C     SCALE TO INTERQUARTILE +(-) INTERQUARTILE DISTANCE.
C
   55 II=0
      III=0
      DO 31 I=1,NY
       J=(Y(I)-YMIN)*VSCALE+1.
       IF(Y(I).LT.CLOW) II = II+1
       IF(Y(I).GT.CHI) III=III+1
       IF(J.GT.50 .OR. J.LT.1) GO TO 31
       IF(Y(I).GE.CLOW .AND.Y(I).LT.XLOW) PLOT(IX,J)=CO
       IF(LFLAG .OR. Y(I).LT.XLOW) GO TO 26
C      THIS IS THE LOW-CROSS POINTER (1ST POINT GE XLOW)
       LFLAG=.TRUE.
       ILX=J
C      NEXT LINE ENDS UP WITH HI-CROSS POINTER (LAST POINT LE XHI)
   26  IF(Y(I).LE.XHI) IHX=J
       IF(Y(I).GT.XHI .AND.Y(I).LE.CHI) PLOT(IX,J)=CO
   31 CONTINUE
C
C     PRINT NUMBER OF OUTLIERS UNLESS 0.
      DO 22 K=1,2
      IK=II
      J=(CLOW-YMIN)*VSCALE + 1
      IF(J.LT.0) J=1
      IF(K.EQ.2) IK=III
      IF(K.EQ.2) J=(CHI-YMIN)*VSCALE + 1
      IF(J.GT.50) J=50
      IF (IK.EQ.0) GO TO 22
      CALL NUMPRT(IX,J,IK,PLOT)
   22 CONTINUE
C
C     FILL BARS ABOVE AND BELOW THE BOX
   56 DO 32 I=ILX,IQ1
       PLOT(IX,I)=CBAR
   32 CONTINUE
      DO 33 I=IQ3,IHX
```

```
      FLOT(IX,I)=CBAR
   33 CONTINUE
      FLOT(IX,ILX)=CROSS
      FLOT(IX,IHX)=CROSS
      FLOT(IX,IQ1)=DASH
      FLOT(IX,IQ2)=CROSS
      FLOT(IX,IQ3)=DASH
      SUM=0.
      DO 40 I=1,NY
       SUM=SUM+Y(I)
   40 CONTINUE
      SUM=SUM/FLOAT(NY)
      MEAN=(SUM-YMIN)*VSCALE+1
      FLOT(IX,MEAN)=CSTR
   99 CONTINUE
      RETURN
C****************************************************************************
      ENTRY SCALE(ULH,DLH)
C     COMPUTES X,Y SCALE AND LIMITS
      XMIN=ULH(1)
      XMAX=ULH(3)
      YMIN=ULH(2)
      YMAX=ULH(4)
      HSCALE=(DLH(3)-DLH(1))/(ULH(3)-ULH(1))
      VSCALE=(DLH(4)-DLH(2))/(ULH(4)-ULH(2))
      RETURN
      END
C****************************************************************************
C
      SUBROUTINE FREG(XS,YS,BS,M,N,IX1,IX2)
C  ROBUST REGRESSION ON Y=X*B
C  X=M BY N MATRIX CONTAINED IN AN ARRAY OF DIM(IX1,IX2)
C  Y=M-VECTOR CONTAINED IN AN ARRAY OF DIM(IX1)
C  B=N-VECTOR CONTAINED IN AN ARRAY OF DIM(IX2)
C  XX,XXI=WORK ARRAYS OF DIM(IX2,IX2)
C  WY=WORK ARRAY OF DIM(IX1)
C  WX=WORK MATRIX OF DIM(IX1,IX2)
C  XY=WORK ARRAY OF DIM(IX2)
C  WK=WORK ARRAY OF DIM(N**2 + 3*N)     OR LARGER
C ***** +++++ +++++  MODIFICATION  USING CHOLESKY +++++
C
      REAL*4 YS(8),XS(8,4),BS(4)
      REAL*8 Y(8),X(8,4),B(4),XTX(4,4),XTY(4)
C ***** CONVERT  REAL*4 TO REAL*8 ****************************
      DO 10 I=1,IX1
         Y(I)=DBLE(YS(I))
         DO 5 J=1,IX2
            X(I,J)=DBLE(XS(I,J))
```

```
  5      CONTINUE
 10   CCNTINUE
C
      CALL MATSQ  (X,XTX,M,N)
      CALL MATMUL (X,Y,XTY,M,N)
      CALL CHOLES (XTX,XTY,B,N)
C
C ***** CONVERT  REAL*8 TO REAL*4 ****************************
      DC 15 J=1,IX2
         BS(J)=SNGL(B(J))
 15   CCNTINUE
C
      RETURN
      END
C
C*****************************************************************
C
      SUBROUTINE NUMPRT(IX,J,IK,PLOT)
C
C     NUMPRT PLOTS THE NUMBER IK IN THE 2-D ARRAY PLOT CENTERED ON
C     THE PLOT(IX,J) POSITION.
C     IX = CCLUMN OF MATRIX PLOT WHERE NUMBER IS TO BE PRINTEC.
C     J = ROW OF MATRIX WHERE NUMBER IS TO BE PRINTED.
C     IK = NUMBER TO BE PRINTED
C     PLOT = 2-D ARRAY WHERE NUMBER IS TO BE PLOTED.
C
      INTEGER*2 NUM(10),PLCT(122,50)
      DATA NUM/'0','1','2','3','4','5','6','7','8','9'/
      IF (IK.LT.10) GO TO 1
      IF (IK.LT.100) GO TO 2
      IF (IK.LT.1000) GC TO 3
      IF (IK.LT.10000) GO TO 4
      I10000 = IK/10000
      PLOT(IX-2,J) = NUM(I10000+1)
      I1000 = (IK-I10000*10000)/1000
      PLOT(IX-1,J) = NUM(I1000+1)
      I100 = (IK-I10000*10000-I1000*1000)/100
      PLOT(IX,J) = NUM(I100+1)
      I10 = (IK-I10000*10000-I1000*1000-I100*100)/10
      PLCT(IX+1,J) = NUM(I10+1)
      I1 = (IK-I10000*10000-I1000*1000-I100*100-I10*10)
      PLCT(IX+2,J) = NUM(I1+1)
      GC TO 22
    4 I1000 = IK/1000
      PLCT(IX-2,J) = NUM(I1000+1)
      I100 = (IK-I1000*1000)/100
      PLCT(IX-1,J) = NUM(I100+1)
      I10 = (IK-I1000*1000-I100*100)/10
```

```
      PLOT(IX,J)  = NUM(I10+1)
      I1 =  (IK-I1000*1000-I100*100-I10*10)
      PLOT(IX+1,J) = NUM(I1+1)
      GO TO 22
    3 I100 = IK/100
      PLOT(IX-1,J) = NUM(I100+1)
      I10 =  (IK-I100*100)/10
      PLOT(IX,J)  = NUM(I10+1)
      I1 =  (IK-I100*100-I10*10)
      PLOT(IX+1,J) = NUM(I1+1)
      GO TO 22
    2 I10 = IK/10
      PLOT(IX-1,J) = NUM(I10+1)
      I1 =  (IK-I10*10)
      PLOT(IX,J)  = NUM(I1+1)
      GO TO 22
    1 PLOT(IX,J)  = NUM(IK+1)
   22 RETURN
      END
C
C*************************************************************************
C
      SUBROUTINE SECEST(X,N,M,NEK,EST,Y,KP,IR,IRK)
      REAL X(IR,IRK),Y(12500),XW(5000,3)
C
C     COMPUTE ESTIMATES "EST" FOR SECTION LENGTH NEK
C     *** IRK2 MUST MATCH THE SECOND DIMENSION OF XW ***
      DATA IRK2/ 3/
      DATA IDR /5000/
C
      IF (IRK.LE.IRK2) GO TO 20
         WRITE (6,2)
    2    FORMAT (1X,'*** ERROR IN SUBROUTINE SECEST ***')
         WRITE (6,3)
    3    FORMAT (1X,'SECOND DIMENSION OF ARRAY XW DOES NOT MATCH')
         WRITE (6,4)
    4    FORMAT (1X,'CHANGE DIMENSION  A N D  VALUE OF IRK2')
      STOP
   20 CONTINUE
C
      NBK=N/NEK
      KE=0
      DO 100 I=1,M
       IE=(I-1)*N + 1
       IE2=IP+NEK
       DO 95 J=1,NBK
        KP=KE+1
        KK=0
```

```
          DO 60 II=IP,IP2
            KK=KK+1
            DO 40 IIRK=1,IRK
              XW(KK,IIRK)=X(II,IIRK)
   40         CONTINUE
   60       CONTINUE
          Y(KP)=EST(XW,NEK,IDR,IRK)
          IP=IP+NEK
          IP2=IP2+NEK
   95   CONTINUE
  100 CONTINUE
      RETURN
      END
C
C*************************************************************************
C
      SUBROUTINE MAXMIN(Y,N,YMAX,YMIN)
C     RETURNS MAX AND MIN VALUES OF VECTOR Y OF LENGTH N
      REAL Y(N)
      YMAX=Y(1)
      YMIN=Y(1)
      DO 605 J=1,N
       IF(Y(J).LT. YMIN) YMIN=Y(J)
       IF(Y(J).GT. YMAX) YMAX=Y(J)
  605 CONTINUE
      RETURN
      END
C
C*************************************************************************
C
      FUNCTION PCTL(Y,N,P)
C     COMPUTES P PERCENTILE OF N VALUES IN Y
      REAL Y(N)
      R=P*FLOAT(N+1)
      CALL SORT(Y,N)
      I=MAX0(INT(R) , 1)
      I=MIN0(I,N)
      J=MIN0(INT(R+1.)  , N)
      R=R-INT(R)
      PCTL=Y(I)+R*(Y(J)-Y(I))
      RETURN
      END
C*************************************************************************
      SUBROUTINE DELETO(Y,KP,YMAX,YMIN)
C     SUBROUTINE SCALES THE GRAPH TO UPPER (LOWER) QUARTILE + (-)
C     1.5 TIMES INTERQUARTILE DISTANCE OR TO FIRST POINT WITHIN
C     THESE LIMITS IF NO POINTS EXIST OUTSIDE.
      REAL Y(KP),Z(12500)
```

```
C       CCPY BEFORE SORTING
        DO 23 I=1,KP
          Z(I) = Y(I)
   23   CONTINUE
        P25 =PCTL(Z,KP,.25)
        P75 =PCTL(Z,KP,.75)
        P50 =PCTL(Z,KP,.50)
        YMIN=2.5*P25-1.5*P75
        YMAX=2.5*P75-1.5*P25
        IF(Z(1).GT.YMIN)  YMIN=Z(1)
        IF(Z(KP).LT.YMAX)  YMAX=Z(KP)
        RETURN
        END
C*****************************************************************
C       CHOLESKI'S  METHODE  *************************
        SUBROUTINE CHOLES (XTX,XTY,BHAT,N)
        REAL*8 L(4,4),SUM,LT(4,4),XTX(4,4),XTY(4),BHAT(4),WY(4)
        REAL*4 B(4)
        INTEGER P
C
C ***** INIT L ******************
        DO 100 I=1,N
          BHAT(I)=0.0D0
          DO 50 J=1,N
           L(I,J)=0.0D0
           LT(I,J)=0.0D0
   50     CONTINUE
  100   CONTINUE
C ****   ALGORITHM  DECOMPOSITION   *********
        L(1,1)=DSQRT(XTX(1,1))
        DO 500 K=2,N
          KK=K-1
          DO 200 J=1,KK
           JJ=J-1
           SUM=0.0D0
           IF  (J.EQ.1)  GO TO 150
            DO  140  P=1,JJ
             SUM=SUM+(L(K,P)*L(J,P))
  140       CONTINUE
  150      CONTINUE
           L(K,J)=(XTX(K,J)-SUM)/L(J,J)
  200     CONTINUE
          SUM=0.0D0
          DO 300 P=1,KK
           SUM=SUM+(L(K,P)**2)
  300     CONTINUE
          L(K,K)= DSQRT (XTX(K,K)-SUM)
  500   CONTINUE
```

```
C  BUILD L-TRANSPOSE  IN  LT *********************
      DO 540 I=1, N
       DO 530 J=1, N
        LT(I,J)=L(J,I)
 530    CONTINUE
 540  CONTINUE
C
C *****  A L G O R I T H M   PART 1 A. 2 ***********
C ***     L * WY = XTY
      WY(1)=XTY(1)/L(1, 1)
      DO 700 I=2,N
         II=I-1
         SUM=0.0DC
         DO 600 J=1,II
            SUM=SUM+(WY(J)*L(I,J))
 600     CONTINUE
         WY(I)=(XTY(I)-SUM)/L(I,I)
 700  CONTINUE
C
C ***   LT * BHAT = WY   *******************
      BHAT(N)=WY(N)/LT(N,N)
      DO 800 II=2,N
         I=N-II+1
         SUM=0.0DC
         DO 750 J=I,N
            SUM=SUM+(BHAT(J)*LT(I,J))
 750     CONTINUE
         BHAT(I)=(WY(I)-SUM)/LT(I,I)
 800  CONTINUE
C
      DO 950 I=1,4
         B(I)=SNGL(BHAT(I))
 950  CONTINUE
C
      RETURN
      END
C
C *****   MATRIX MULTIPLICATION  XT * X = XRES  *******
C
      SUBROUTINE MATSQ ( X, XRES, M, N )
      REAL*8 X(8,4),XT(4,8),XRES(4,4),SUM
C
C ***   BUILD  X-TRANSPOSE IN LT  **********
      DO 20 I=1,M
         DO 10 J=1,N
            XT(J,I)=X(I,J)
 10      CONTINUE
 20   CONTINUE
```

```
C
C ***    XT * X = XRES    ****
C
      DO 50  I=1,N
         DO 40  J=1,N
            SUM=0.0D0
            DO 3C  K=1,M
               SUM=SUM+(XT(I,K)*X(K,J))
 30            CONTINUE
               XRES (I,J)  =  SUM
 40      CONTINUE
 50   CONTINUE
      RETURN
      END
C
C  ****  MATRIX MULTIPLICATION  XT * Y = XTY  ********
C
      SUBROUTINE MATMUL  ( X,Y,XTY,M,N )
      REAL*8 Y(8),XT(4,8),X(8,4),XTY(4),SUM
C
C ****   BUILD XT   ******
      DO 20  I=1,M
         DO  10  J=1,N
            XT(J,I)=X(I,J)
 10      CONTINUE
 20   CONTINUE
C
C ****   XT * Y = XTY   *******
C
      DO 50  I=1,N
         SUM=0.0D0
         DO 40  J=1,M
            SUM=SUM+ (XT(I,J)*Y(J))
 40      CONTINUE
         XTY(I)=SUM
 50   CONTINUE
      RETURN
      END
C
C ******************************************************************
C
      SUBROUTINE SORT (Y,N)
C   INPLACE SORT USING SHELL ALGORITHM ********
         REAL Y(N),TEMP
      INTEGER GAP
      LOGICAL EXCH
C
      GAP= (N/2)
```

```
    5   IF (.NOT.(GAP.NE.0)) GO TO 500
   10      CONTINUE
              EXCH=.TRUE.
              K=N-GAP
              DO 200 I=1,K
              KK=I+GAP
              IF(.NOT.(Y(I).GT.Y(KK))) GO TO 100
                 TEMP=Y(I)
                 Y(I)=Y(KK)
                    Y(KK)=TEMP
                    EXCH=.FALSE.
  100         CONTINUE
  200         CONTINUE
           IF (.NOT.(EXCH)) GO TO 10
           GAP=(GAP/2)
        GO TO 5
  500   CONTINUE
C
        RETURN
        END
```

APPENDIX C

SIMTB3 (VERSION 3) PROGRAM LISTING

```
C
C                 SIMTB3  PROGRAM  LISTING
C
C
C
C
C  PURPOSE      TO GENERATE REGRESSION ADJUSTED ESTIMATES AND BOX PLOTS
C               OF ESTIMATES OF AN INPUT RAW DATA SERIES X CONTAING M
C               (REPLICATIONS) OF N VALUES EACH.  UP TO 3 ESTIMATING
C               FUNCTIONS CAN BE USED.  THE GRAPHS CAN ALL BE OF THE SAME
C               SCALE OR SCALED INDIVIDUALLY.
C  *            THE SERIES X IS GENERATED FROM INSIDE SIMTB3 BY USER
C  *            PROVIDED FUNCTION
C
C     DESCRIPTION OF PARAMETERS
C
C *   GEND1, GEND2, GEND3: USER FUNCTIONS THAT WILL GENERATE THE DATA.
C *             THE NAMES MUST BE DECLARED AS EXTERNAL IN THE
C *             CALLING PROGRAM.  THEIR CALLS MUST BE AS FOLLOWS:
C *                 CALL FUNCTION_NAME(IX,X,NX)
C *             WHERE IX IS THE SEED, X THE ARRAY & NX THE NO.TO GENERATE
C
C *   ISEED1, ISEED2, ISEED3:  SEEDS FOR DATA GENERATORS 1, 2 & 3.
C *             THE SEEDS ARE UPDATED (ADVANCED) UPON RETURN FROM SIMTB3
C
C *   Y         WORK ARRAY OF SIZE >= M*N/NE(1))
C
C     N         NUMBER OF DATA ELEMENTS PER SECTION (N IS SAMPLE SIZE).
C
C     M         NUMBER OF SECTIONS (REPLICATIONS).
C               M CANNOT EXCEED 100
C
C     NE        INTEGER ARRAY OF SIZE 8 CONTAINING SUBSAMPLE SIZES FOR N.
C               THE VALUES OF NE MUST BE FROM SMALLEST TO LARGEST.
C               NO ELEMENT OF THE ARRAY NE CAN BE GREATER THAN N.
C
C     L         NUMBER OF SUBSAMPLE SIZES FROM NE(8) THAT WILL BE USED TO
C               SECTION N.
C               IT IS ALSO THE NUMBER OF BOXPLOTS THAT WILL BE PRODUCED.
C
C     D         DEGREE OF REGRESSION FOR MEAN AND VARIANCE REGRESSIONS.
C               D WILL BE REDUCED BY SIMTB3 IF THE SAMPLE IS NOT LARGE
C               ENOUGH. D MUST BE 1,2 OR 3. D=0 WILL IGNORE REGRESSIONS.
C
C               *** SCALING ***
C                 SCALING IS ACCOMPLISHED BY TAKING THE SMALLEST AND THE
C                 LARGEST ESTIMATE VALUES FROM ALL ESTIMATING FUNCTIONS
C                 EVALUATED ON THE SHORTEST SECTION LENGTH ( NE(1) ).
C                    THE SEI PARAMETER ALLOWS THE USER TO SCALE THE GRAPHS
```

```
C              CF EACH ESTIMATOR INDIVIDUALLY CR TC SCALE THEM ALL TO
C              THE SAME SCALE.  SCALING ALL TO THE SAME SCALE IS
C              ACCOMPLISHED BY TAKING THE MINIMUM ANC MAXIMUM  ESTIMATE
C              FROM ALL THE ESTIMATORS USING NE(1) SUBSAMPLE SIZE.
C                 THE RG PARAMETER ALLOWS THE USER TO RREDUCE THE VER-
C              TICAL SCALE TO: THE UPPER QUARTILE DISTANCE + 1.5 TIMES
C              INTERQUARTILE DISTANCE AS THE MAX VALUE AND THE LOWER
C              QUARTILE - 1.5 TIMES THE INTERQUARTILE DISTANCE AS THE
C              MIN VALUE.  THE INTERQUARTILE DISTANCE IS COMPUTED FROM
C              THE SAMPLE OF ESTIMATES FROM THE NE(1) SUBSAMPLE SIZE.
C              IF THERE ARE NO ESTIMATES OUTSIDE THESE MIN AND MAX
C              VALUES THEN THE SCALE IS TO THE FIRST VALUE WITHIN.
C              IF THERE ARE ESTIMATES OUTSIDE THESE LIMITS THEN THEY
C              ARE COUNTED AND THE NUMBER PRINTED AT THE ENDS OF THE
C              BOX PLOTS.
C                 THE SVS PARAMET ALLOWS THE USER TO SET THE VERTICAL
C              SCALE. WHEN THE VERTICAL SCALE IS SET THE SEI PARAMETER
C              IS IGNORED AND THE VERTICAL SCALE BECOMES YMIN AND YMAX.
C
C
C     RG       RG=0 DO NCT REDUCE THE VERTICAL SCALE OF THE GRAPHS.
C              RG=1 REDUCE GRAPHICS VERTICAL SCALE TC UPPER (LOWER)
C                   QUARTILE + (-) INTERQUARTILE DISTANCE.
C
C     SEI      SEI=0 DO NOT SCALE ESTIMATORS' GRAPHS INDIVIDUALLY.
C              SEI=1 SCALE ESTIMATORS' GRAPHS INDIVIDUALLY.
C
C     SVS      SVS=0  PROGRAM WILL CALCULATE VERTICAL SCALE.
C              SVS=1  USER SETS VERTICAL SCALE TC YMIN AND YMAX.
C
C     YMIN     LOW VALUE OF VERTICAL SCALE. SET BY USER WHEN SVS=1
C
C     YMAX     HIGH VALUE OF VERTICAL SCALE. SET BY USER WHEN SVS=1
C
C     NEST     NUMBER OF ESTIMATORS THAT WILL BE USED TO CALCULATE
C              STATISTICAL PARAMETER FROM X DATA.
C              NEST MUST BE 1,2 OR 3.
C
C     EST1     NAMES OF THE ESTIMATOR FUNCTIONS THAT WILL BE USED TO
C     EST2     CALCULATE THE STATISTICAL PARAMETER.
C     EST3     CALL SEQUENCE ON EACH FUNCTION IS: CALL FNAME(X,N) WHERE
C              X IS THE DATA ARRAY AND N IS THE NUMBER OF DATA POINTS.
C              THEY MUST BE DECLARED IN THE CALLING PROGRAM IN
C              THE ORDER THEY ARE USED. DUMMY VARIABLES MUST BE INSERTED
C              WHEN THERE ARE LESS THAN 3 ESTIMATORS.
C
C     TTL1     TITLES ASSOCIATED WITH EACH ESTIMATOR (EST1,2,3).  A MAX
C     TTL2     OF 120 CHARACTERS CAN BE USED TO DESCRIBE EACH ESTIMATOR.
```

```
C     TTL3     EACH TITLE MUST BE DECLARED AS REAL*8(15) ARRAYS UNLESS
C              PASSED BY VALUE (WHEN PASSING THE TITLE BY VALUE THERE
C              MUST BE A MINIMUM OF 120 CHARS. BETWEEN APOSTROPHES.
C
C*****************************************************************
C
      SUBROUTINE SIMTB3(ISEED1,ISEED2,ISEED3,Y,N,M,NE,L,D,RG,SEI,SVS,
     *YMIN,YMAX,NEST, GEND1,EST1,TTL1, GEND2,EST2,TTL2,GEND3,EST3,TTL3)
C
      REAL ULH(4),Y(20000),GV(2)
      REAL*8 TTL1(15),TTL2(15),TTL3(15)
      INTEGER NE(8),RG,SEI,SVS,SM
      INTEGER D,I,NEST,TEST
C
      GV(1)=1.E30
      GV(2)=-1.E30
      SM=NE(1)
      MN = M*N
      LT=L-1
      IF(LT.EQ.0) GO TO 13
      DO 11 I=1,LT
         I1=I+1
         IF(NE(I).GT.NE(I1)) WRITE(6,110)
   11 CONTINUE
   13 TEST=0
      IF(NEST.EQ.1 .OR. NEST.EQ.2 .OR. NEST.EQ.3) GO TO 2
       WRITE(6,106)
       TEST=1
    2 IF(M.GE.1.AND.M.LE.100) GO TO 3
       WRITE(6,104)
       TEST=1
    3 IF(L.GE.1.AND.L.LE.8) GO TO 4
       WRITE(6,103)
       TEST=1
    4 IF(D.LE.3) GO TO 5
       WRITE(6,108)
       TEST=1
    5 CONTINUE
      IF(NE(L) .LT. N) GO TO 6
       WRITE(6,107)
       TEST=1
    6 CONTINUE
      IF (TEST.NE.0) GO TO 80
       ULH(2)=YMIN
       ULH(4)=YMAX
C     DETERMINE HOW EACH GRAPH IS TO BE SCALED.
      IF (SVS .EQ. 1) GO TO 50
      IF (SEI .EQ. 1) GO TO 75
```

```
C
C     ****************************************************************
C     *GRAPH ALL ESTIMATORS TO THE SAME SCALE OF ESTIMATOR W/WIDEST PTS.
C     ****************************************************************
      ULH(2)=1.E30
      ULH(4)=-1.E30
      ISEED=ISEED1
      DO 10 IK=1,1
C      FIND VERTICAL SCALE FOR 1ST ESTIMATOR.
       CALL SECEST(GEND1,ISEED,N,M,NE(IK),EST1,Y,KP)
        IF(RG.EQ.1) CALL DELETO(Y,KP,YMAX,YMIN)
        IF(RG.NE.1) CALL MAXMIN(Y,KP,YMAX,YMIN)
        IF(YMIN .LT. ULH(2)) ULH(2)=YMIN
        IF(YMAX .GT. ULH(4)) ULH(4)=YMAX
   10 CONTINUE
C
C     FIND VERTICAL SCALE FOR 2ND ESTIMATOR.  KEEP WIDEST PAIR****
      IF(NEST .LT. 2) GO TO 40
      ISEED=ISEED2
      DO 20 IK=1,1
       CALL SECEST(GEND2,ISEED,N,M,NE(IK),EST2 ,Y,KP)
        IF(RG.EQ.1) CALL DELETO(Y,KP,YMAX,YMIN)
        IF(RG.NE.1) CALL MAXMIN(Y,KP,YMAX,YMIN)
        IF(YMIN .LT. ULH(2)) ULH(2)=YMIN
        IF(YMAX .GT. ULH(4)) ULH(4)=YMAX
   20 CONTINUE
C
C     FIND VERTICAL SCALE FOR 3RD ESTIMATOR.  KEEP WIDEST PAIR****
      IF(NEST .LT. 3) GO TO 40
      ISEED=ISEED3
      DO 30 IK=1,1
       CALL SECEST(GEND3,ISEED,N,M,NE(IK),EST3 ,Y,KP)
        IF(RG.EQ.1) CALL DELETO(Y,KP,YMAX,YMIN)
        IF(RG.NE.1) CALL MAXMIN(Y,KP,YMAX,YMIN)
        IF(YMIN .LT. ULH(2)) ULH(2)=YMIN
        IF(YMAX .GT. ULH(4)) ULH(4)=YMAX
   30 CONTINUE
C     RETURN CALCULATED SCALE TO CALLER
   40 CONTINUE
      YMIN=ULH(2)
      YMAX=ULH(4)
C
C     PROCESS BOXPLOTS USING FIXED VERTICAL SCALE IN VECTOR ULH
C     ONE CALL FOR EACH ESTIMATOR USED.
   50 CONTINUE
      CALL PRST(GEND1,ISEED1,N,M,EST1,NE,L,RG,L,ULH,Y,GV)
      WRITE(6,101) ULH(2),ULH(4)
      WRITE(6,102) TTL1
```

```
      IF (NEST. LT. 2) GC TO 80
      CALL PRST(GEND2,ISEED2,N,M,EST2,NE,L,RG,D,ULH,Y,GV)
      WRITE(6,101) ULH(2),ULH(4)
      WRITE(6,102) TTL2
      IF (NEST. LT. 3) GO TO 80
      CALL PRST(GEND3,ISEED3,N,M,EST3,NE,L,RG,D,ULH,Y,GV)
      WRITE(6,101) ULH(2),ULH(4)
      WRITE(6,102) TTL3
      GC TO 80
C     ****************************************************
C     *GRAPH EACH ESTIMATOR SCALED TO ITS WIDEST POINTS.*
C     ****************************************************
C     FIND VERTICAL SCALE FOR 1ST ESTIMATOR AND GRAPH.
   75 CCNTINUE
      ISEED=ISEED1
      CALL SECEST(GEND1,ISEED,N,M,NE(1),EST1,Y,KP)
      IF(RG.EQ.1) CALL DELETO(Y,KP,YMAX,YMIN)
      IF(RG.NE.1) CALL MAXMIN(Y,KP,YMAX,YMIN)
      ULH(2)=YMIN
      ULH(4)=YMAX
      CALL PRST(GEND1,ISEED1,N,M,EST1,NE,L,RG,D,ULH,Y,GV)
      WRITE(6,101) ULH(2),ULH(4)
      WRITE(6,102) TTL1
      IF(NEST .LT. 2) GC TO 80
C
C     FIND VERTICAL SCALE FCR 2ND ESTIMATOR AND GRAPH.
      ISEED=ISEED2
      CALL SECEST(GEND2,ISEED,N,M,NE(1),EST2 ,Y,KP)
      IF(RG.EQ.1) CALL DELETO(Y,KP,YMAX,YMIN)
      IF(RG.NE.1) CALL MAXMIN(Y,KP,YMAX,YMIN)
      ULH(2)=YMIN
      ULH(4)=YMAX
      CALL PRST(GEND2,ISEED2,N,M,EST2,NE,L,RG,D,ULH,Y,GV)
      WRITE(6,101) ULH(2),ULH(4)
      WRITE(6,102) TTL2
      IF(NEST .LT. 3) GC TO 80
C
C     FIND VERTICAL SCALE FCR 3RD ESTIMATOR AND GRAPH.
      IF (SVS.EQ.1) GO TO 78
      ISEED=ISEED3
      CALL SECEST(GEND3,ISEED,N,M,NE(1),EST3 ,Y,KP)
      IF(RG.EQ.1) CALL DELETO(Y,KP,YMAX,YMIN)
      IF(RG.NE.1) CALL MAXMIN(Y,KP,YMAX,YMIN)
      ULH(2)=YMIN
      ULH(4)=YMAX
   78 CALL PRST(GEND3,ISEED3,N,M,EST3,NE,L,RG,D,ULH,Y,GV)
      WRITE(6,101) ULH(2),ULH(4)
      WRITE(6,102) TTL3
```

```
C
   80 CONTINUE
      WRITE(6,122) GV
  122 FORMAT(' *** WIDEST Y VALUES FOUND:  YMIN=',G10.4,
     *       '          YMAX=',G10.4)
  102 FORMAT(1X,'ESTIMATOR: ',15A8)
  101 FORMAT(1X,'VERTICAL SCALE:  YMIN =',F10.4,/,18X,'YMAX =',F10.4/)
  103 FORMAT(' *** ERROR...L MUST BE AN INTEGER BETWEEN 1 AND 8. ***')
  104 FORMAT(' *** ERROR...M MUST BE AN INTEGER BETWEEN 1 AND 100. ***')
  106 FORMAT(' *** ERROR...NEST MUST BE 3 OR LESS. ***')
  107 FORMAT(' *** ERROR...N/NE(L) MUST BE 1 OR GREATER TO COMPUTE',
     +' STATISTICS.',/,'    INCREASE N OR DECREASE NE(L). ***')
  108 FORMAT(' *** ERROR...D MUST BE LESS THAN OR EQUAL TO 3. ***')
  110 FORMAT(' *** WARNING... NE ARRAY ELLEMENTS ARE NOT IN ORDER OF '/
     +'      INCREASING SIZE. IF NE(1) IS NOT SMALLEST ELEMENT, SCALING'/
     +'      MAY CAUSE POINTS TO FALL OUTSIDE RANGE OF SCALE.')
      RETURN
      END
C
C***********************************************************************
C
      SUBROUTINE ERST(GENDAT,ISEED,N,M,EST,NE,L,RG,UD,ULH,Y,GV)
C     REGRESSION ADJUSTED ESTIMATE
C     CALCULATES ESTIMATES FROM USER DATA USING "EST" FUNCTION
C     PLOTS BASIC OR RETRENCHED GRAPH ON LINE PRINTER
C  *  GENDAT = SUBROUTINE NAME TO GENERATE THE DATA
C
C  *  ISEED = SEED FOR GENDAT
C
C     EST = NAME OF USER WRITTEN ESTIMATING FUNCTION.
C           USAGE:      FUNCTION NAME(X,N)
C           WHERE X IS A VECTOR WITH N ENTRIES
C
C     M = NO. OF REPLICATIONS (MUST BE <= 100)
C     N = NUMBER OF VALUES IN EACH REPLICATION (M*N MUST BE <= 50000)
C     X = USERS VECTOR WITH M CONSECUTIVE BATCHES OF N VALUES EACH
C     L = NO. OF SECTION SIZES (MUST BE BETWEEN 1 AND 8)
C     NE = ARRAY WITH THE L SUBSAMPLE SIZES (MUST BE IN ASCENDING ORDER)
C     UD = DEGREE OF THE REGRESSION (MUST BE <= 3 & <= L-1 )
C     ULH= XMIN,YMIN,XMAX,YMAX  IN USER UNITS
C          ONLY ULH(2) AND ULH(4) NEED TO BE PASSED. OTHERS CALC. HERE
C     GV=WIDEST Y VALUES. PRINTED ON LAST PLOT TO AID IN SETTING SCALE
C
      DIMENSION  NE(L)
      INTEGER NB(8),LOCX(8),D1,IWIDTH,UD,D,LT,DT,RG,RNEK
      INTEGER*2 PLOT(122,50), CBAR,BLK,DASH,CSTR,NUM(10),DOT
      REAL*4 ULH(4),DLH(4),Y(20000),GV(2)
      REAL*4 RH(8,100),STAT(8,6),VT(8)
```

```
      REAL*8 SUM2,SUM3,SUM4,LABEL(5)
      REAL*4 RA(8,4),RV(8,4),B(4,100),V(4),BA(4),EV(4),BS(4),RT(8),BT(8)
C
      DATA DLH/1.,1.,122.,50./
      DATA BLK/' '/, DASH/'-'/, CBAR/'|'/, DOT/'.'/
      DATA LABEL/'MEAN','STD','STD MEAN','SKEWNESS','KURTOSIS'/
      D=MINO(3,UL,I-1)
      IX1=8
      IX2=4
      MN=M*N
      D1=D+1
      IWIDTH= IPIX(DLH(3))
C     BUILD REGRESSION MATRICES FOR AVERAGES AND VARIANCES
      DO 84 K=1,I
       DO 86 J=1,D1
        T=FLOAT(NE(L))/FLOAT(NE(K))
        RA(K,J)=T**(J-1)
        RV(K,J)=T**((FLOAT(J))/2.0)
   86  CONTINUE
   84 CONTINUE
C     CLEAR PLOT ARRAY
      DO 3 J=1,50
       DO 4 I=1,122
        PLOT(I,J)=BLK
    4  CONTINUE
    3 CONTINUE
C     SET HORIZONTAL XMIN, XMAX
      ULH(1)=.7*NE(1)
      ULH(3)=1.2*NE(L)
C     SET SCALE
      CALL SCALE(ULH,DLH)
C     COMPUTE LOCATION OF BOXPLOTS ALONG X-AXIS
      LAST=-1
      DO 5 K=1,L
       NB(K)=N/NE(K)
       LOCX(K)=(NE(K)-ULH(1))*(DLH(3)-DLH(1))/(ULH(3)-ULH(1))+DLH(1)+.5
       IF(LOCX(K).LT. LAST+4) LOCX(K)=LAST+4
       LAST=LOCX(K)
    5 CONTINUE
C
      DO 80 K=1,I
       NEK=NB(K)
       FNEK=NE(K)
C      SECTION & COMPUTE ESTIMATORS FOR SIZE K
       CALL SECEST(GENDAT,ISEED,N,M,RNEK,EST,Y,KF)
C      AVERAGE ESTIMATES OF SIZE NE(K) FOR EACH OF M REPLICATIONS
       KF=0
       DO 10 I=1,M
```

```
      RH(K,I)=0.
      DO 15 J=1,NBK
       KP=KP+1
       GV(1)=AMIN1(GV(1), Y(KP))
       GV(2)=AMAX1(GV(2), Y(KP))
       RH(K,I)=RH(K,I)+Y(KP)
15    CONTINUE
      RH(K,I)=RH(K,I)/FLOAT(NBK)
10   CONTINUE
     CALL BOXPRT(Y,KP,LOCX(K),PLOT,RG)
     IF (K .GT. 8) GO TO 80
C    COMPUTE MEAN AND MOMENT ESTIMATES
     XMEAN=0.
     DO 180 IM1=1,KP
      XMEAN=XMEAN+Y(IM1)
180  CONTINUE
     XMEAN=XMEAN/FLOAT(KP)
     SUM2 = 0.D0
     SUM3 = 0.D0
     SUM4 = 0.D0
     DO 190 IP1=1,KP
      DEV = Y(IP1) - XMEAN
      SUM2 = SUM2 + DEV * DEV
      SUM3 = SUM3 + DEV ** 3
      SUM4 = SUM4 + DEV ** 4
190  CONTINUE
C
C    CHECK TO INSURE SAMPLE SIZE IS LARGE ENOUGH FOR
C    EACH MOMENT COMPUTATION.
     IF (KP.LT.2) GO TO 7
     VAR = SUM2 / (KP - 1.0)
     STDV = SQRT(VAR)
7    IF (KP.LT.3) GO TO 8
     XSUM3 = SNGL(SUM3) * KP /( (KP-1.) * (KP-2.))
     SKEW = XSUM3 / STDV ** 3
8    IF (KP.LT.4) GO TO 9
     XSUM4 = SNGL(SUM4)*((KP-2.)*KP+3.)/((KP-1.)*(KP-2.)*(KP-3.))
     XSUM4 = XSUM4 - VAR*VAR*3.*(KP-1.)*(KP+KP-3.)/(KP*(KP-2.)*(KP-3))
     CKURT = XSUM4 / (VAR * VAR) - 3.
9    STAT(K,1)=XMEAN
     STAT(K,2)=STDV
     STAT(K,3)=STDV/SQRT(FLOAT(KP))
     STAT(K,4)=SKEW
     STAT(K,5)=CKURT
     STAT(K,6)=VAR
80  CONTINUE
C
C   IF D1.LT.2 THEN NO REGRESSIONS OR PLOTING CAN BE DONE
```

```
C
      IF(D1.LT.2) GO TO 113
      DO 92 K=1,E
       DO 47 J=1,L
        BT(J)=RH(J,K)
47     CONTINUE
       CALL RREG(EA,RT,BT,L,D1,IX1,IX2)
       E(1,K)=BT(1)
       DO 23 KT=2,D1
        E(KT,K)=E1(KT)*NE(I)**(KT-1)
23     CONTINUE
92    CONTINUE
C
C     AVERAGE REGRESSION COEFF. OVER M REPLICATIONS & CALC. VARIANCE
C
      DO 94 I=1,D1
       BA(I)=0.
       BV(I)=0.
       DO 95 J=1,M
        BA(I)=BA(I)+B(I,J)
        BV(I)=BV(I)+B(I,J)**2
95     CONTINUE
       BA(I)=BA(I)/FLOAT(M)
       IF(M.EQ.1) GO TO 94
       BV(I)=(BV(I)-M*BA(I)**2)/(M*(M-1.))
       ES(I)=BV(I)**.5
94    CONTINUE
C
C     ESTABLISH REGRESSION LINE & ASYMPTOTE
C
      DO 98 I=3,IWIDTH
C      MAP I FROM DEVICE SPACE TO USER SPACE
       UX=(I-DLH(1))*(ULH(3)-ULH(1))/(DLH(3)-DLH(1)) + ULH(1)
C      COMPUTE THE Y VALUE FROM X AND THE REGRESSION COEFFICIENTS.
       UY=BA(1)
       DO 99 J=1,L
        UY=UY+BA(J+1)/UX**J
99     CONTINUE
C
C
C      MAP THE Y VALUE FROM USER SPACE TO DEVICE SPACE
       J=(UY-ULH(2))*(DLH(4)-DLH(2))/(ULH(4)-ULH(2)) + DLH(2) + .5
       IF(J .LT. 1 .OR. J .GT. 50) GO TO 98
        IF(PLOT(I,J) .EQ. BLK) PLOT(I,J)=DOT
98    CONTINUE
C
C      SCALE ASYMPTOTE, BETA0, AND PLOT ACROSS PLOT.
       J=(BA(1)-ULH(2))*(DLH(4)-DLH(2))/(ULH(4)-ULH(2)) + DLH(2) + .5
```

```
      IF(J .LT. 1 .OR. J .GT. 50) GO TO 117
      DO 120 I=3,IWIDTH
       IF(PLOT(I,J) .EQ. BLK) PLOT(I,J)=DASH
  120 CONTINUE
C
C     REGRESSION ON VARIANCES FROM EACH SEGMENT WITH A VARIANCE.
C
  117 K=M*(N/NE(L))
      LT=L
      DT=D1
      DO 111 I=1,L
         IF(K.GE.2) GO TO 112
         LT=LT-1
         K=M*(N/NE(LT))
  111 CONTINUE
  112 IF(LT.LT.DT) DT=LT
      IF(DT.LT.2) GO TO 113
      DO 48 J=1,L
         VT(J)=STAT(J,6)*(NE(J)**0.5)
   48 CONTINUE
      CALL RREG(EV,VT,V,LT,DT,IX1,IX2)
      DO 77 I=1,LT
         V(I) = V(I)*NE(L)**(FLOAT(I)/2.)
   77 CONTINUE
C
C     PLOT    *************
C
  113 WRITE(6,102)
      WRITE(6,161) N,M,D
      WRITE(6,101)
      DO 90 J=1,50
       K=51-J
       IF(MOD(K,5).NE.0) GO TO 85
       YLABEL=(K-DLH(2))*(ULH(4)-ULH(2))/(DLH(4)-DLH(2)) + ULH(2)
       WRITE(6,103)  YLABEL,(PLOT(I,K),I=1,IWIDTH)
       GO TO 90
   85  CONTINUE
       WRITE(6,100)   (PLOT(I,K),I=1,IWIDTH)
   90 CONTINUE
C     LABEL X-AXIS BY REUSING PLOT MATRIX.
      DO 115 I=1,122
        PLOT(I,1)=DASH
        PLOT(I,2)=BLK
  115 CONTINUE
      DO 130 J=1,L
       PLOT(LOCX(J),1)=CBAR
       IK = NE(J)
       IX = LOCX(J)
```

```
      CALL NUMPRT(IX,2,IK,PLOT)
  130 CONTINUE
      WRITE(6,106)  (PLOT(I,1),I=1,IWIDTH)
      WRITE(6,104)  (PLOT(I,2),I=1,IWIDTH)
      L8=L
      IF(L8 .GT. 8) L8=8
      WRITE(6,156)
      WRITE(6,146)  (NE(I),I=1,L8)
      WRITE(6,157)  LABEL(1),  (STAT(K,1),K=1,L8)
C     CHECK TO INSURE SAMPLE SIZE IS LARGE ENOUGH TO COMPUTE
C     STATISTICS BEFORE ATTEMPTING TO PRINT STATS.
      LT=L8
      L1=L8
      DO 21  I=1,LT
         K1  = M*(N/NE(L1))
         IF  (K1.GE.2) GO TO 11
         L1  = L1-1
   21 CONTINUE
      GO TO 14
   11 WRITE(6,157) LABEL(2),  (STAT(K,2),K=1,L1)
      WRITE(6,157) LABEL(3),  (STAT(K,3),K=1,L1)
      LT = L1
      DO 22  I=1,LT
         K1  = M*(N/NE(L1))
         IF  (K1.GE.3) GO TO 12
         L1  = L1-1
   22 CONTINUE
      GO TO 14
   12 WRITE(6,158) LABEL(4),  (STAT(K,4),K=1,L1)
      LT = L1
      DO 33  I=1,LT
         K1  = M*(N/NE(L1))
         IF  (K1.GE.4) GO TO 13
         L1  = L1-1
   33 CONTINUE
      GO TO 14
   13 WRITE(6,158) LABEL(5),  (STAT(K,5),K=1,L1)
      IF(D1.LT.2) GO TO 444
      WRITE(6,151)  (BA(I),I=1,D1)
   14 IF(M.LT.2) GO TO 444
      WRITE(6,152)  (BV(I),I=1,D1)
      WRITE(6,153)  (BS(I),I=1,D1)
  444 IF(DT.LE.1) GO TO 999
      WRITE(6,155)  (V(I),I=1,DT)
  999 WRITE(6,162)
  100 FORMAT(9X,'|',122A1,'|')
  101 FORMAT(9X,'+',122('-'),'+')
  102 FORMAT('1')
```

```
103 FORMAT(' ',G8.2,'-',122A1,'|')
104 FORMAT(10X,122A1)
106 FORMAT(9X,'+',122A1,'+')
107 FORMAT(10G13.8)
151 FORMAT(/,' MEAN OF REGRESSION ON AVERAGES - COEFFICIENTS:',4G20.6)
152 FORMAT(' VARIANCE OF REGRESSION - COEFFICIENTS:',8X,4G20.6)
153 FORMAT(' STD DEV OF REGRESSION - COEFFICIENTS:',9X,4G20.6)
156 FORMAT(/' SUBSAMPLE')
146 FORMAT(' SIZE      ',8(I8,6X),/)
157 FORMAT(1X,A6,8G14.4)
158 FORMAT(1X,A6,8F14.3)
159 FORMAT(/,' REGRESSION ON VARIANCE - COEFFICIENTS: ',7X,4G20.6)
161 FORMAT(9X,'SAMPLE SIZE (N):',I5,20X,'NO. OF REPLICATIONS (M):',
   +I5,18X,'DEGREE OF REGRESSION (D):',I3)
162 FORMAT(/)
    RETURN
    END
C
C*************************************************************************
C
    SUBROUTINE BOXPRT(Y,NY,IX,PLOT,RG)
C   PREPARES BOXPLOT  FROM VECTOR Y (IN 2-D ARRAY PLOT)
C
    REAL Y(NY),ULH(4),DLH(4)
    INTEGER RG
    INTEGER*2 PLOT(122,50),DASH,CBAR,CROSS,CSTR,CC,NUM(10)
    LOGICAL*1 IFLAG
C
    DATA DASH/'-'/,CBAR/'|'/,CSTR/'*'/,CROSS/'+'/,CC/'0'/
    IF(NY .GE. 9) GO TO 5
C   WHEN LESS THAN 9 POINTS JUST SHOW THE POINTS
    DO 8 I=1,NY
     J=(Y(I)-YMIN)*VSCALE + 1.
C    IGNORE VALUE IF IT FALLS OUTSIDE WINDOW
     IF(J.GT.50  .OR.  J.LT.1) GO TO 8
     PLOT(IX,J)=CO
  8 CONTINUE
    SUM=0.
    DO 88 I=1,NY
     SUM=SUM+Y(I)
 88 CONTINUE
    SUM=SUM/FLOAT(NY)
    MEAN=(SUM-YMIN)*VSCALE+1
    PLOT(IX,MEAN)=CSTR
    GO TO 99
  5 CONTINUE
    IFLAG=.FALSE.
    P25 =PCTL(Y,NY,.25,0)
```

```
      P75 =PCTL(Y,NY,.75,1)
      P50 =PCTL(Y,NY,.50,1)
      IC1=(P25-YMIN)*VSCALE+1.
      IC2=(P50-YMIN)*VSCALE+1.
      IC3=(P75-YMIN)*VSCALE+1.
      XLOW=2*P25-P75
      ILOW=(XLOW-YMIN)*VSCALE+1.
      XHI=2*P75-P25
      IHI=(XHI-YMIN)*VSCALE+1.
      CLOW=2.5*P25-1.5*P75
      CHI=2.5*P75-1.5*P25
C     DRAW BOX
      DO 20 I=IQ1,IQ3
        PLOT(IX-1,I)=CBAR
        PLOT(IX+1,I)=CBAR
   20 CONTINUE
      PLOT(IX-1,IC1)=DASH
      PLOT(IX+1,IC1)=DASH
      PLOT(IX-1,IC3)=DASH
      PLOT(IX+1,IC3)=DASH
C     DETERMINE IF OUTLIERS ARE TO BE COUNTED AND THE NUMBER PRINTED.
      IF (RG.EQ.1) GO TO 55
      DO 30 I=1,NY
        J=(Y(I)-YMIN)*VSCALE+1.
C       IGNORE VALUE IF IT FALLS OUTSIDE WINDOW
        IF(J.GT.50  .OR.  J.LT.1)  GO TO 30
        IF(Y(I).LT.CLOW)  PLOT(IX,J)=CSTR
        IF(Y(I).LT.CLOW)  GO TO 30
        IF(Y(I).GE.CLOW  .AND.Y(I).LT.XLOW)  PLOT(IX,J)=CO
        IF(LFLAG  .OR. Y(I).LT.XLOW)  GO TO 25
C       THIS IS THE LOW-CROSS POINTER (1ST POINT GE XLOW)
        LFLAG=.TRUE.
        IIX=J
C       NEXT LINE ENDS UP WITH HI-CROSS POINTER (LAST POINT LE XHI)
   25   IF(Y(I).LE.XHI)  IHX=J
        IF(Y(I).GT.XHI  .AND.Y(I).LE.CHI)  PLOT(IX,J)=CO
        IF(Y(I).GT.CHI)  PLOT(IX,J)=CSTR
   30 CONTINUE
      GO TO 56
C
C     SCALE TO INTERQUARTILE +(-) INTERQUARTILE DISTANCE.
C
   55 II=0
      III=0
      DO 31 I=1,NY
        J=(Y(I)-YMIN)*VSCALE+1.
        IF(Y(I).LT.CLOW)  II = II+1
        IF(Y(I).GT.CHI)  III=III+1
```

```
      IF(J.GT.50  .OR.  J.LT.1) GO TO 31
      IF(Y(I).GE.CLOW .AND.Y(I).LT.XLOW) PLOT(IX,J)=CO
      IF(LFLAG .OR. Y(I).LT.XLOW) GO TO 26
C     THIS IS THE LOW-CROSS POINTER (1ST POINT GE XLOW)
      LFLAG=.TRUE.
      ILX=J
C     NEXT LINE ENDS UP WITH HI-CROSS POINTER (LAST POINT LE XHI)
   26 IF(Y(I).LE.XHI) IHX=J
      IF(Y(I).GT.XHI .AND.Y(I).LE.CHI) PLOT(IX,J)=CO
   31 CONTINUE
C
C     PRINT NUMBER OF OUTLIERS UNLESS 0.
      DO 22 K=1,2
      IK=II
      J=(CLOW-YMIN)*VSCALE + 1
      IF(J.LT.0) J=1
      IF(K.EQ.2) IK=III
      IF(K.EQ.2) J=(CHI-YMIN)*VSCALE + 1
      IF(J.GT.50) J=50
      IF (IK.EQ.0) GO TO 22
      CALL NUMPRT(IX,J,IK,PLOT)
   22 CONTINUE
C
C     FILL BARS ABOVE AND BELOW THE BOX
   56 DO 32 I=ILX,IQ1
       PLOT(IX,I)=CBAR
   32 CONTINUE
      DO 33 I=IQ3,IHX
       PLOT(IX,I)=CBAR
   33 CONTINUE
      PLOT(IX,ILX)=CROSS
      PLOT(IX,IHX)=CROSS
      PLOT(IX,IQ1)=DASH
      PLOT(IX,IQ2)=CROSS
      PLOT(IX,IQ3)=DASH
      SUM=0.
      DO 40 I=1,NY
       SUM=SUM+Y(I)
   40 CONTINUE
      SUM=SUM/FLOAT(NY)
      MEAN=(SUM-YMIN)*VSCALE+1
      PLOT(IX,MEAN)=CSTR
   99 CONTINUE
      RETURN
C*********************************************************************
      ENTRY SCALI(ULH,DLH)
C     COMPUTES X,Y SCALE AND LIMITS
      XMIN=ULH(1)
```

```
      XMAX=ULH(3)
      YMIN=ULH(2)
      YMAX=ULH(4)
      HSCALE=(DLH(3)-DLH(1))/(ULH(3)-ULH(1))
      VSCALE=(DLH(4)-DLH(2))/(ULH(4)-ULH(2))
      RETURN
      END
C
C*************************************************************************
C
      SUBROUTINE FREG(XS,YS,BS,M,N,IX1,IX2)
C     ROBUST REGRESSION ON Y=X*B
C     X=M BY N MATRIX CONTAINED IN AN ARRAY OF DIM(IX1,IX2)
C     Y=M-VECTOR CONTAINED IN AN ARRAY OF DIM(IX1)
C     B=N-VECTOR CONTAINED IN AN ARRAY OF DIM(IX2)
C     XX,XXI=WORK ARRAYS OF DIM(IX2,IX2)
C     WY=WORK ARRAY OF DIM(IX1)
C     WX=WORK MATRIX OF DIM(IX1,IX2)
C     XY=WORK ARRAY OF DIM(IX2)
C     WK=WORK ARRAY OF DIM(N**2 + 3*N)     OR LARGER
C
      REAL*4 YS(8),XS(8,4),BS(4)
      REAL*8 Y(8),X(8,4),B(4),XTX(4,4),XTY(4)
C***** CONVERT  REAL*4 TO REAL*8 ****
      DO 10 I=1,IX1
         Y(I)=DBLE(YS(I))
         DO 5 J=1,IX2
            X(I,J)=DBLE(XS(I,J))
  5      CONTINUE
 10   CONTINUE
C
      CALL MATSQ  (X,XTX,M,N)
      CALL MATMUI  (X,Y,XTY,M,N)
      CALL CHOLES  (XTX,XTY,B,N)
C
C***** CONVERT  REAL*8 TO REAL*4 ****
         DO 15 J=1,IX2
            BS(J)=SNGL(B(J))
 15      CONTINUE
C
      RETURN
      END
C*************************************************************************
      SUBROUTINE NUMPRT(IX,J,IK,PLOT)
C
C     NUMPRT PLOTS THE NUMBER IK IN THE 2-D ARRAY PLOT CENTERED ON
C     THE PLOT(IX,J) POSITION.
C     IX = COLUMN OF MATRIX PLOT WHERE NUMBER IS TO BE PRINTED.
```

```
C       J = ROW OF MATRIX WHERE NUMBER IS TO BE PRINTED.
C       IK = NUMBER TO BE PRINTED
C       PLOT = 2-D ARRAY WHERE NUMBER IS TO BE PLOTED.
C
        INTEGER*2 NUM(10) ,PLOT(122,50)
        DATA NUM/'0','1','2','3','4','5','6','7','8','9'/
        IF (IK.LT. 10) GO TO 1
        IF (IK.LT. 100) GO TO 2
        IF (IK.LT. 1000) GO TO 3
        IF (IK.LT. 10000) GO TO 4
        I10000 = IK/10000
        PLOT(IX-2,J) = NUM(I10000+1)
        I1000 = (IK-I10000*10000)/1000
        PLOT(IX-1,J) = NUM(I1000+1)
        I100 = (IK-I10000*10000-I1000*1000) /100
        PLOT(IX,J) = NUM(I100+1)
        I10 = (IK-I10000*10000-I1000*1000-I100*100)/10
        PLOT(IX+1,J) = NUM(I10+1)
        I1 = (IK-I10000*10000-I1000*1000-I100*100-I10*10)
        PLOT(IX+2,J) = NUM(I1+1)
        GO TO 22
      4 I1000 = IK/1000
        PLOT(IX-2,J) = NUM(I1000+1)
        I100 = (IK-I1000*1000)/100
        PLOT(IX-1,J) = NUM(I100+1)
        I10 = (IK-I1000*1000-I100*100) /10
        PLOT(IX,J) = NUM(I10+1)
        I1 = (IK-I1000*1000-I100*100-I10*10)
        PLOT(IX+1,J) = NUM(I1+1)
        GO TO 22
      3 I100 = IK/100
        PLOT(IX-1,J) = NUM(I100+1)
        I10 = (IK-I100*100)/10
        PLOT(IX,J) = NUM(I10+1)
        I1 = (IK-I100*100-I10*10)
        PLOT(IX+1,J) = NUM(I1+1)
        GO TO 22
      2 I10 = IK/10
        PLOT(IX-1,J) = NUM(I10+1)
        I1 = (IK-I10*10)
        PLOT(IX,J) = NUM(I1+1)
        GO TO 22
      1 PLOT(IX,J) = NUM(IK+1)
     22 RETURN
        END
C*************************************************************************
C
        SUBROUTINE SECEST(GENDAT,IX,N,M,NEK,EST,Y,KP)
```

```
      REAL X(2000),Y(1)
      INTEGER IX
C     COMPUTE ESTIMATES "EST" FOR SECTION LENGTH NEK (LIMITED TO 2000)
      NBK=N/NEK
      KP=0
      DO 10 I=1,M
       DO 15 J=1,NBK
        KP=KP+1
        CALL GENDAT(IX,X,NEK)
        Y(KP)=EST(X,NEK)
   15  CONTINUE
   10 CONTINUE
      RETURN
      END
C
C************************************************************************
C
      SUBROUTINE MAXMIN(Y,N,YMAX,YMIN)
C     RETURNS MAX AND MIN VALUES OF VECTOR Y OF LENGTH N
      REAL Y(N)
      YMAX=Y(1)
      YMIN=Y(1)
      DO 605 J=1,N
       IF(Y(J).LT. YMIN) YMIN=Y(J)
       IF(Y(J).GT. YMAX) YMAX=Y(J)
  605 CONTINUE
      RETURN
      END
C************************************************************************
C
      FUNCTION PCTL(Y,N,P,IC)
C     COMPUTES P PERCENTILE OF N VALUES IN Y.
C     WHEN IC=1 DATA IS ALREADY SORTED
      REAL Y(N)
      R=P*FLOAT(N+1)
      IF(IC .NE. 1) CALL SORT(Y,N)
      I=MAX0(INT(R) , 1)
      I=MIN0(I,N)
      J=MIN0(INT(R+1.) , N)
      R=R-INT(R)
      PCTL=Y(I)+R*(Y(J)-Y(I))
      RETURN
      END
C
C************************************************************************
C
      SUBROUTINE DELETO(Y,KP,YMAX,YMIN)
C     SUBROUTINE SCALES THE GRAPH TO UPPER (LOWER) QUARTILE + (-)
```

```
C     1.5 TIMES INTERQUARTILE DISTANCE OR TO FIRST POINT WITHIN
C     THESE LIMITS IF NO POINTS EXIST OUTSIDE.
      REAL Y(KP)
      P25 =PCTL(Y,KP,.25,0)
      P75 =PCTL(Y,KP,.75,1)
      P50 =PCTL(Y,KP,.50,1)
      YMIN=2.5*P25-1.5*P75
      YMAX=2.5*P75-1.5*P25
      IF(Y(1).GT.YMIN)  YMIN=Y(1)
      IF(Y(KP).LT.YMAX)  YMAX=Y(KP)
      RETURN
      END
C
C*********************************************************************
C
      SUBROUTINE CHOLES (XTX,XTY,BHAT,N)
      REAL*8 L(4,4),SUM,LT(4,4),XTX(4,4),XTY(4),BHAT(4),WY(4)
      REAL*4 B(4)
      INTEGER P
C
C****** INIT L ******************
      DO 100 I=1,N
       BHAT(I)=0.0D0
       DO 50 J=1,N
        L(I,J)=0.0D0
        LT(I,J)=0.0D0
  50   CONTINUE
 100  CONTINUE
C*****  ALGORITHM  DECOMPOSITION  *********
      L(1,1)=DSQRT(XTX(1,1))
      DO 500 K=2,N
       KK=K-1
       DO 200 J=1,KK
        JJ=J-1
        SUM=0.0D0
        IF  (J.EQ.1)  GO TO 150
         DO  140  P=1,JJ
          SUM=SUM+(L(K,P)*L(J,P))
 140     CONTINUE
 150    CONTINUE
        L(K,J)=(XTX(K,J)-SUM)/L(J,J)
 200   CONTINUE
       SUM=0.0D0
       DO 300 P=1,KK
        SUM=SUM+(L(K,P)**2)
 300   CONTINUE
       L(K,K)=  DSQRT  (XTX(K,K)-SUM)
 500  CONTINUE
```

```
C  BUILD L-TRANSPOSE  IN  IT *********************
      DO 540 I=1, N
       DO 530 J=1, N
        LT(I,J)=L(J,I)
 530    CONTINUE
 540  CONTINUE
C
C******  A L G O R I T H M   PART 1 A. 2 **********
C ***      L * WY = XTY
      WY(1)=XTY(1)/L(1, 1)
      DO 700 I=2,N
         II=I-1
         SUM=0.0DC
         DO 600 J=1,II
            SUM=SUM+ (WY(J)*L(I,J))
 600     CONTINUE
         WY(I)= (XTY(I)-SUM)/L(I,I)
 700  CONTINUE
C
C ***   LT * BHAT = WY   ****
      BHAT(N)=WY(N)/LT(N,N)
      DO 800 II=2,N
         I=N-II+1
         SUM=0.0DC
         DO 750 J=I,N
            SUM=SUM+ (BHAT(J)*LT(I,J))
 750     CONTINUE
         BHAT(I)=(WY(I)-SUM)/LT(I,I)
 800  CONTINUE
C
      DO 950 I=1,4
         B(I)=SNGL(BHAT(I))
 950  CONTINUE
C
      RETURN
      END
C
C******   MATRIX MULTIPLICATION  XT * X = XRES  *******
C
      SUBROUTINE MATSQ ( X, XRES, M, N )
      REAL*8 X(8,4),XT(4,8),XRES(4,4),SUM
C
C ***   BUILD  X-TRANSPOSE IN LT  **********
C
      DO 20  I=1,M
         DO 10 J=1,N
            XT(J,I)=X(I,J)
 10      CONTINUE
```

```
 20   CONTINUE
C
C*****    XT * X = XRES    ****
C
      DO 50 I=1,N
          DO 40 J=1,N
              SUM=0.0D0
              DO 30 K=1,M
                  SUM=SUM+(XT(I,K)*X(K,J))
 30               CONTINUE
                  XRES (I,J) = SUM
 40       CONTINUE
 50   CONTINUE
      RETURN
      END
C
C  ****  MATRIX MULTIPLICATION  XT * Y = XTY  ********
C
      SUBROUTINE MATMUL ( X,Y,XTY,M,N )
      REAL*8 Y(8),XT(4,8),X(8,4),XTY(4),SUM
C
C*****  BUILD XT  *****
C
      DO 20 I=1,M
          DO 10 J=1,N
              XT(J,I)=X(I,J)
 10       CONTINUE
 20   CONTINUE
C
C*****  XT * Y = XTY  ****
C
      DO 50 I=1,N
          SUM=0.0D0
          DO 40 J=1,M
              SUM=SUM+(XT(I,J)*Y(J))
 40       CONTINUE
          XTY(I)=SUM
 50   CONTINUE
      RETURN
      END
C
C***************************************************************************
C
      SUBROUTINE SORT (Y,N)
C   INPLACE SORT USING SHELL ALGORITHM ********
          REAL Y(N),TEMP
      INTEGER GAP
      LOGICAL EXCH
```

```
C
      GAP=(N/2)
    5 IF (.NOT.(GAP.NE.0)) GO TO 500
   10    CONTINUE
            EXCH=.TRUE.
            K=N-GAP
            DO 200 I=1,K
            KK=I+GAP
            IF(.NOT.(Y(I).GT.Y(KK))) GO TO 100
               TEMP=Y(I)
               Y(I)=Y(KK)
                  Y(KK)=TEMP
                  EXCH=.FALSE.
  100       CONTINUE
  200       CONTINUE
         IF (.NOT.(EXCH)) GO TO 10
         GAP=(GAP/2)
      GO TO 5
  500 CONTINUE
C
      RETURN
      END
```

LIST OF REFERENCES

1. Heidelberger, P. and Lewis, P.A.W. (1981). "Regression-Adjusted Estimates for Regenerative Simulations, with Graphics," Communications of the ACM, v. 24, pp. 260-273.

2. Linnebur, D.G. (1982). A Graphical Testbed for Analyzing and Reporting the Results of a Simulation Experiment, Master's Thesis, Naval Postgraduate School, Monterey, California.

3. Anderson, T.W. and Walker, A.M. (1964). "On the Asymptotic Distribution of the Autocorrelation of a Sample from a Linear Stochastic Process," Ann. Math. Statist., 35, pp. 1296-1303.

4. Cox, D.R. (1966). "The Null Distribution of the First Serial Correlation Coefficient," Biometrika, 53, pp. 523-626.

5. Mosteller, F. and Tukey, J.W. (1977). Data Analysis and Regression, Addison-Wesley.

6. Cramer, H. (1946). Mathematical Methods of Statistics, Princeton University Press, Princeton, New Jersey.

7. Cox, D.R. and Lewis, P.A.W. (1981), The Statistical Analysis of Series of Events, Chapman and Hall, London.

8. Johnson, N.L. and Kotz, S. (1970), Continuous Univariate Distributions--1, Houghton Mifflin Company, Boston.

9. Efron, B. and Gong, G. (1983). "A Leisurely Look at the Bootstrap, the Jackknife, and Cross-Validation," The American Statistician, 1, pp. 36-48.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93943	2
3.	Department Chairman, Code 55 Department of Operations Research Naval Postgraduate School Monterey, California 93943	1
4.	Professor P.A.W. Lewis, Code 55Lw Naval Postgraduate School Monterey, California 93943	10
5.	Professor P.A. Jacobs, Code 55Jc Naval Postgraduate School Monterey, California 93943	1
6.	Professor E.J. Orav, Code 55 Naval Postgraduate School Monterey, California 93943	2
7.	Major Hans-Walter Drueg German Military Representative USA/CA - Sl - Washington, D.C. 20016-1887	2

END

FILMED

2-84

DTIC